UROP: Undergraduate Research Opportunity Program
Completion Certificate

Name: Seung Bin, Joo

Duration: 9 weeks, 2022.07.12 ~ 2022.09.08

Research Topic: Pneumatic Control Strategy to Reduce Landing Impact and Vibration in Human Walking

This certificate is presented to Seung Bin, Joo for completing the Undergraduate Research Opportunity Program (UROP) in the BioRobotics Laboratory, Seoul National University

8th of September, 2022

Seoul National University,
Mechanical Engineering,
BioRobotics Lab

Professor. Kyu Jin Cho (印)

SNU Biorobotics Laboratory UROP

Seung-Bin Joo

Pre-lab reading:

- Pneumatic actuators convert compressed air into mechanical motion (linear or rotary)
- Basics of pneumatics, another video
- PneuNets (design, fabrication, testing, modelling)
- Pouch motors
- “Pouch Motors: Printable/Inflatable Soft Actuators for Robotics”
- “A Fully Fabric-Based Bidirectional Soft Robotic Glove for Assistance and Rehabilitation of Hand Impaired Patients”
- “A Control and Drive System for Pneumatic Soft Robots: PneuSoRD”
- Air suspension
- “An investigation into improved air bag performance by vent control and gas injection”

13/07/2022:

Things I can do

- Look at pneumatic pumps (we want high pressure and high flow rate, but the pump must be compact)
- Find out how solenoid valves work
- Learn how to construct pneumatic systems (pumps, tubes, valves, actuator)
- Learn how to use MyRIO LabVIEW with pneumatic pumps, actuator, tubes, solenoid valve, pressure sensor, etc
- Basic pneumatic circuits

14/07/2022:

Basic exploration of pneumatics with arduinos:

- Arduino UNO and cable
- Pneumatic tubed
- Resistors and potentiometers
- Op-amps
- Pneumatic fitting
- Pneumatic pump
- Pressure sensors
- Solenoid valve
- Breadboard
- 12V supply
- Transistors
- Diodes
- Soldering
- Motor controller

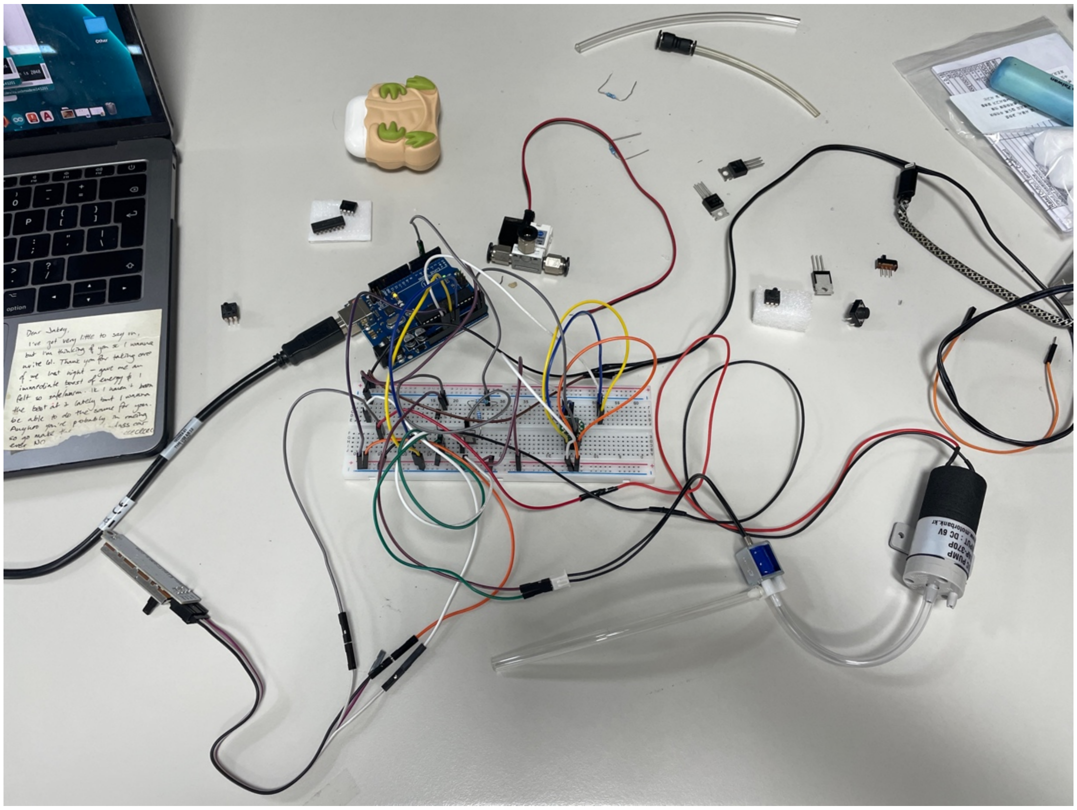
- Pressure sensor (xgzp with amplifier)
- Switch
- Voltage regulator (can be used as transistor)
- Potentiometers

Questions:

- Do I need a relay for operating the 6V DC pump? Where do I find the relay?
- Can I operate 6V pump with 12V adaptor?
- Can you operate the pump with a potential divider circuit (instead of the relay)? How? → Use linear potentiometer to change voltage delivered to the pump
- Where can I find transistors? Specific type of transistor needed for the solenoid valve?
- Where can I find diodes?
- The plastic housing on the pump? Can I connect that using wires? Struggles with wire connections in general.
- Pressure sensor? All online documents are about pressure sensors with breakout boards? → Use pressure sensor with op amps (bigger op amp is essentially 4 op amps in one)
- What value resistors to use?

Resources used:

- https://www.google.com/search?q=breadboard+layout&sxsrf=ALiCzsZKwsCfGGmG-nGTbRgN_lva-9c13w:1657762055251&source=lnms&tbm=isch&sa=X&ved=2ahUKEwj2j7PznPf4AhUfmVYBHW3zCFgQ_AUoAXoECAEQAw&biw=1440&bih=789&dpr=2#imgrc=vE_IfOsq-FNmSM
- https://arduinogetstarted.com/tutorials/arduino-relay
- https://arduinogetstarted.com/tutorials/arduino-controls-pump
- https://makersportal.com/blog/2020/6/4/mps20n0040d-pressure-sensor-calibration-with-arduino
- https://www.teachmemicro.com/arduino-pressure-sensor-tutorial/
- https://create.arduino.cc/projecthub/sarful/digital-pressure-sensor-arduino-workshop-0e43ae
- https://bc-robotics.com/tutorials/controlling-a-solenoid-valve-with-arduino/
- https://www.google.com/search?q=how+to+read+resistor+sizes&sxsrf=ALiCzsasUQ8CU0HRoUIUvpKTwBAVAmS0bg:1657767133498&source=lnms&tbm=isch&sa=X&ved=2ahUKEwj73fLor_f4AhWpmlYBHcEeClYQ_AUoAXoECAEQAw&biw=1440&bih=789&dpr=2#imgrc=59dcJkTY-0wFZM
- https://circuitdigest.com/electronic-circuits/potential-voltage-divider-circuit-diagram
- https://www.youtube.com/watch?v=PUte1cmJ44A
- https://www.instructables.com/Wiring-Linear-Sliding-Potentiometer-With-Arduino/
- https://create.arduino.cc/projecthub/glowascii/transistors-for-robotics-arduino-basics-2cf124
- https://www.instructables.com/How-to-Read-MPX5010-Differential-Pressure-Sensor-W/
- https://www.learnrobotics.org/blog/read-analog-sensors-arduino/
- http://m.eleparts.co.kr/data/goods_old/data/XGZP_SOP6.pdf
- https://docs.arduino.cc/software/ide-v2/tutorials/ide-v2-serial-monitor



```
int pressure = A4;
int solenoidPin = 4;
int threshold = 121;

void pressurecheck(){
int data = analogRead(pressure); // set variable data as analog pressure values read from the pressure pin
  if (data <= threshold){
    digitalWrite(solenoidPin, HIGH); // Switch solenoid on
    delay(1000); // wait one second

    Serial.print("pressure = "); // print "pressure =" on the serial monitor
    Serial.println(data); // print data vertically in the serial monitor

    digitalWrite(solenoidPin, LOW); // Switch solenoid off
    delay(1000); // wait one second
  }
  else{
    Serial.println("End");
    delay(10);
    exit(0);
  }
}

void setup() {
pinMode(pressure, INPUT); // initialize pressure sensor as input
Serial.begin(9600); // begin serial monitor and baud rate 9600

pinMode(solenoidPin, OUTPUT); // set the solenoid pin as an output

}

void loop() {
pressurecheck();

}
```

Air pump operates and its strength is determined by the position of the knob on the linear potentiometer. The solenoid valve is programmed to turn on and off every second. This continues until the pressure sensor reaches a threshold value and then the system stops.

Note: Look at the pressure sensor specification (0 – 40kPa is the range). The readings from the Arduino (120, 121, etc.) must be scaled into the 0 – 40kPa range.

<u>15/07/2022:</u>

Basic plan going forward:

1. Download LabVIEW and myRIO toolkit and learn the environment
2. Learn how to control the 8 solenoid valves that will be used to power the dynamic cushion (paying particular attention to the aperture size and response speed)
3. Apply basic control method (e.g. if we set the pressure to 100kPa in the cushion, how fast can we reach that pressure, how accurately, does it overshoot?)
4. Apply more complex control strategies

Basic information:

- We want to attach a dynamic cushion system to the bottom of the shoe to reduce ground reaction force (GRF)
- We also want to create propulsion using the dynamic cushion to maintain efficiency of walking/running
- Dynamic cushion can be modelled with stiffness and damping
- Valves will input air into the dynamic cushion and air can/will also be expelled
- I can 3D print parts that I feel like I will need (maybe to secure pneumatic tubes or something)
- 8 solenoid valves will be connected to the dynamic cushion

Pouch motor manufacturing process:

1. Take TPU covered fabric (TPU is the slippery side that can melt)
2. Use the laser cutter to cut it into the desired shape (you can input autocad files)
3. Put the blue polymer sheet (Teflon?) in the desired shape between the TPU layers (this is the part that won't fuse)
4. Use the heat press at 170°C and fuse the layers together.

<u>To Do List:</u>

- Learn the LabVIEW and myRIO toolkit environment (computer given to me on Tuesday July 19th) – there is more software that needs to be download along with this, look into it
- Read and fully understand "A Control and Drive System for Pneumatic Soft Robots: PneuSoRD"
- Look at the printed circuit board design from the paper above
- Learn how to use Autodesk Eagle
- Research different type of valves (2-way solenoid valves, proportional valves) → research how to make electrically stable circuits with these valves (capacitors, etc.)

- Research battery packs and how to make the dynamic cushion setup portable

Proportional valves - a proportional valve provides a change in output pressure or flow in the same ratio as the change in the input, for example if the input doubles then the output will also double.

Electronic system

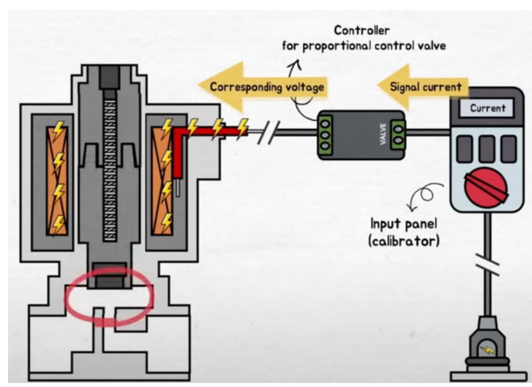
Open loop control system - The open-loop configuration does not monitor or measure the condition of its output signal as there is no feedback

Closed loop control system - Closed-loop Systems use feedback where a portion of the output signal is fed back to the input to reduce errors and improve stability

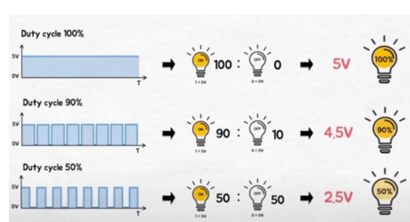
18/07/2022:

Proportional valve:

- Proportional valve can be used to control the flow or pressure in proportion to the change in input voltage



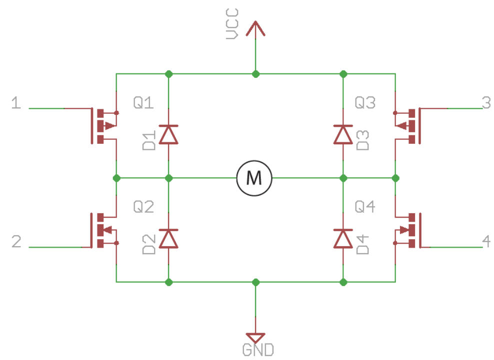
- System usually consists of input panel (calibrator), controller and the proportional valve
- For example, if the controller takes a current between 4 and 20mA, then when the controller receives 20mA it will apply 24V such that the proportional valve orifice fully opens, and maximum flow is achieved.
- If the flow is to be halved, signal current should be 12mA and then the controller will apply 12V so that the orifice is half-open and produces half the amount of flow.
- So how does the proportional valve controller work?
    - The controller receives a signal current and outputs a desired voltage
    - Controller uses pulse width modulation (PWM) control to convert voltage into a mean voltage corresponding to the input signal current



    - PWM is a way of using digital signals to obtain a similar effect as analog signals
    - By adjusting the duty cycle (fraction of a period where the signal is on), we can create different mean voltages.
    - Controller outputs PWM signal to the proportional valve

Motor drivers and controllers:

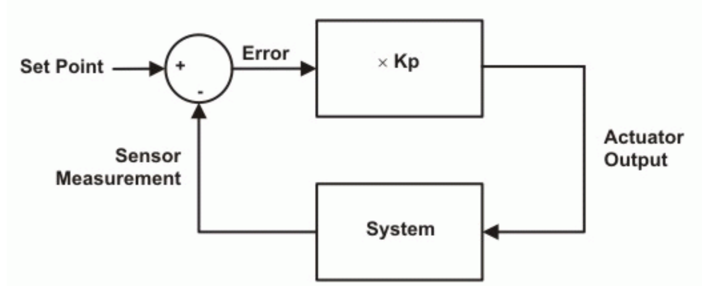
- A microcontroller like an Arduino cannot drive a motor because its pin output a maximum current of 25mA (not even close to what is needed to drive motors)
- Motor drivers accepts small current to then control motors that need large currents using a circuit of MOSFETS (transistors) → they are often on an integrated circuit
- A motor controller is a motor driver with a control interface stacked on top of it
- A motor driver simply handles the power to drive the motors, whereas the logic and digital control has to be done by an external microcontroller or microprocessor, whereas a motor controller has all of the logic circuitry built in and can be controlled by a higher-level interface such as a PWM signal, USB, analogue input etc.
- H bridge MOSFET circuit: turning on the pairs of transistors (2/3 and 1/4) switches the direction of current and hence the direction of the motor
- The speed of the motor can be controlled by a PWM signal



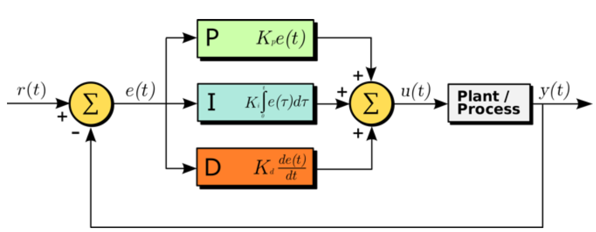
PID control:

- Proportional control loop: difference between set point and sensor measurement is the error term and this error term is multiplied by a constant K_p which then determines the actuator input. The system approaches the set point asymptotically (for a large error term, actuator output is large; for a small error term actuator output is small)

Figure 2 - Proportional Control Block Diagram



- This kind of control loop produces a problem if we for example want a drone to hover at a certain height (50m)
  - At 0m, the error is large, so drone propeller speed is high and the drone moves up
  - At 50m, the error is zero, so drone propeller speed is zero and hence the drone falls
  - Where the drone hovers depends on the gain. For example, say the drone hovers at propeller speed 100 rpm. If the gain is 5, the drone hovers when the error term is 20, which is when height is 30m. In fact, to hover at 50m, we need infinite gain (for finite gains, there will always be an error called steady state error).
- How do we get rid of steady-state error?
  - Add integrator path alongside proportional path



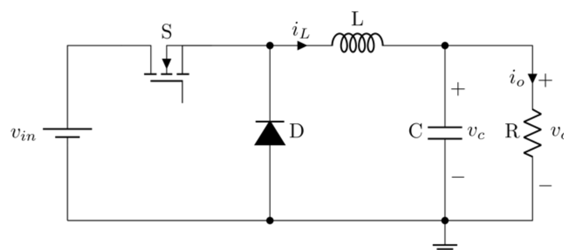
- The integrator sums of the input signal over time and hence has a "memory" of the past
- Say the gain is 5 and the drone hovers at 30m with a steady state error of 20m.
- If a constant steady state error term is integrated, we get a linearly increasing output → increasing propeller speed (actuator output)
- This happens until the error term is zero at which point the integral outputs zero and the drone hovers at 50m.

- While the system tries to reach its desired state (drone at 50m), it may overshoot
  - We can account for this by adding a path that takes the "future" into consideration
  - The derivative path measures how fast the error is changing (growing/shrinking)
  - For example, if the error term is quickly decreasing, the derivative will be a negative value → this negative value is added to the controller output and lowers the propeller speed → this prematurely slows down the propeller speed
- Each branch has its own gain → adjusting these gains is called tuning the controller
- All this combined is called a PID controller

Bang bang controller:

- Bang-bang control is a type of control system that mechanically or electronically turns something on or off when a desired target (setpoint) has been reached.

Buck converter:



- Just like transformers step-down AC voltage, DC voltage is stepped down using a buck converter
- Buck converters step down voltage by increasing the current
- Transistor S receives a PWM signal at its gate which opens and closes the switch, turning v_{in} on and off → if duty cycle of the PWM signal is 40%, then the input voltage is stepped down to 40% of its original value
- When the switch is closed and current flows from the source to the load R, the current goes across an inductor. The inductor opposes the change in current (Lenz's law) so initially the current is low. Eventually, maximum current in the inductor is reached and energy is stored in the inductor in the form of a magnetic field. When the switch is open, inductor continues to drive current to the load (using its stored magnetic energy) until its energy runs out → this causes an accumulation of electronics near transistor S → the diode D creates a path for the electronics to prevent damage
- The capacitor in parallel with the load also helps smoothen the current across the load as the PWM signal is applied (when the switch is opened, capacitor continues to supply load with current)
- Beginning with the switch open (off-state), the current in the circuit is zero. When the switch is first closed (on-state), the current will begin to increase, and the inductor will produce an opposing voltage across its terminals in response to the changing current. This voltage drop counteracts the voltage of the source and therefore reduces the net voltage across the load.

Over time, the rate of change of current decreases, and the voltage across the inductor also then decreases, increasing the voltage at the load. During this time, the inductor stores energy in the form of a magnetic field.

- If the switch is opened while the current is still changing, then there will always be a voltage drop across the inductor, so the net voltage at the load will always be less than the input voltage source. When the switch is opened again (off-state), the voltage source will be removed from the circuit, and the current will decrease. The decreasing current will produce a voltage drop across the inductor (opposite to the drop at on-state), and now the inductor becomes a current source. The stored energy in the inductor's magnetic field supports the current flow through the load. This current, flowing while the input voltage source is disconnected, when appended to the current flowing during on-state, totals to current greater than the average input current (being zero during off-state).

Circuit schematics

19/07/2022:

Goals:

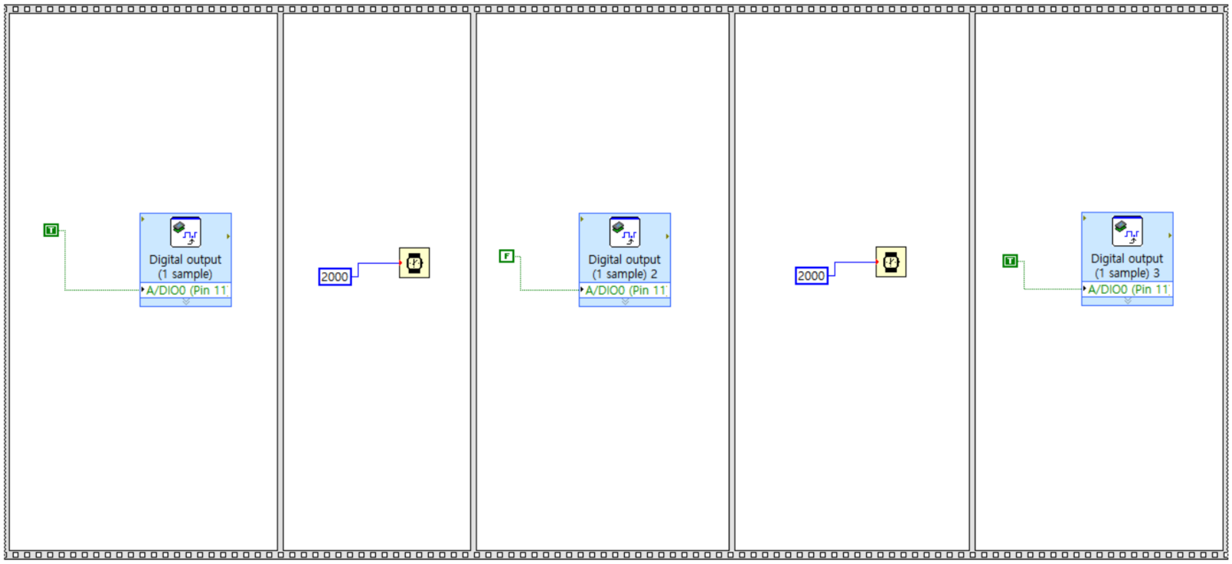
- Install myRIO toolkit and get LabVIEW to work with the myRIO controller
- Learn how to connect circuits with myRIO controller
- Successfully connect circuit for controlling a basic pneumatic system
- Successfully program something on LabVIEW which controls the pneumatic system

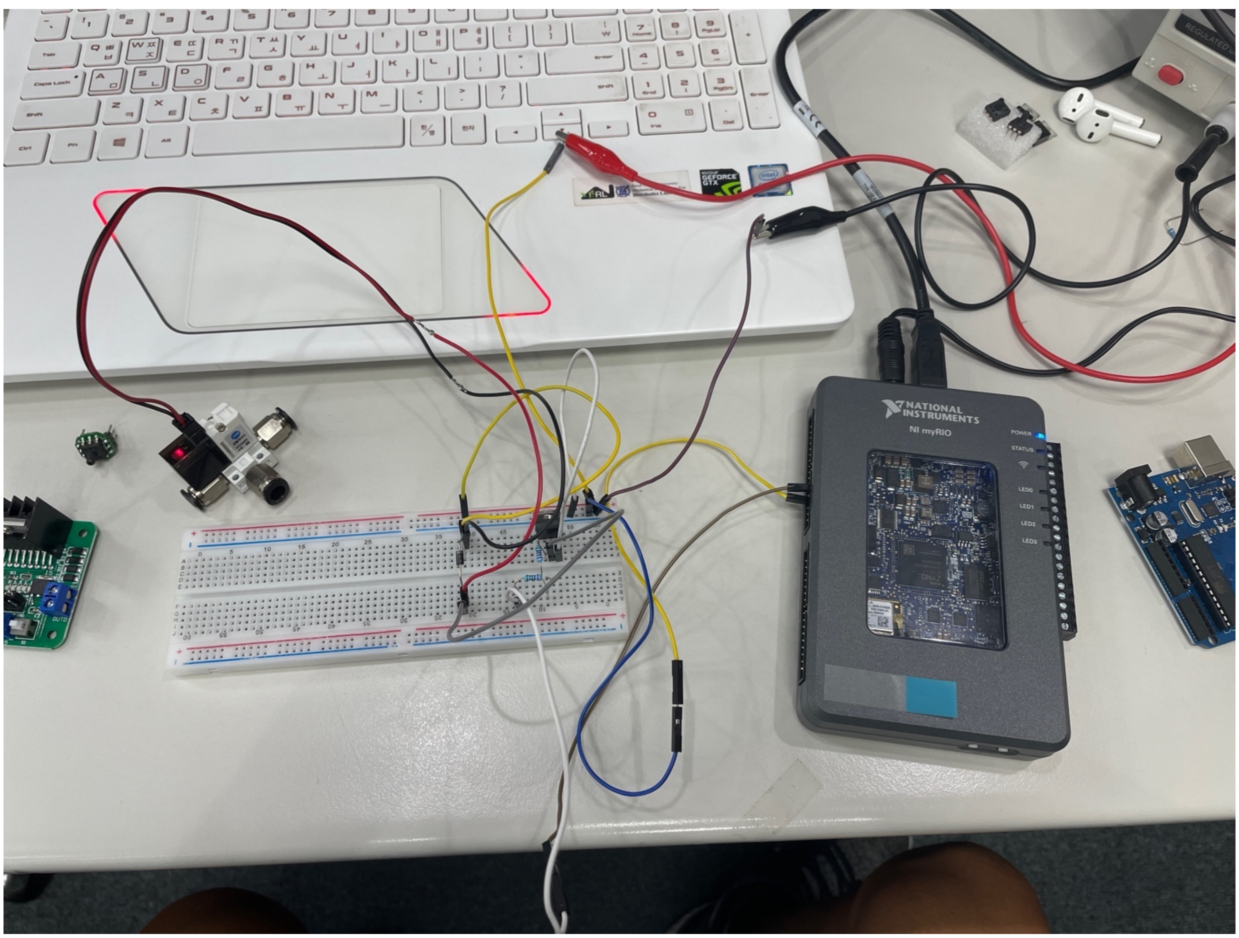
To Do list:

- Draw a pneumatic circuit which inflates and deflates a pneumatic actuator (pouch motor) upon clicking a button (using solenoid valves, pneumatic actuators, tubes, pumps)
- Think about how pressure sensor could be integrated into the pneumatic circuit and how it would be programmed (e.g. if a certain pressure is reached, stop inflating)
- Figure how to control solenoid valves with LabVIEW myRIO (if a certain pressure is set, using a PID control loop to reach that pressure by pumping air)

Basic circuit for connecting solenoid valve:

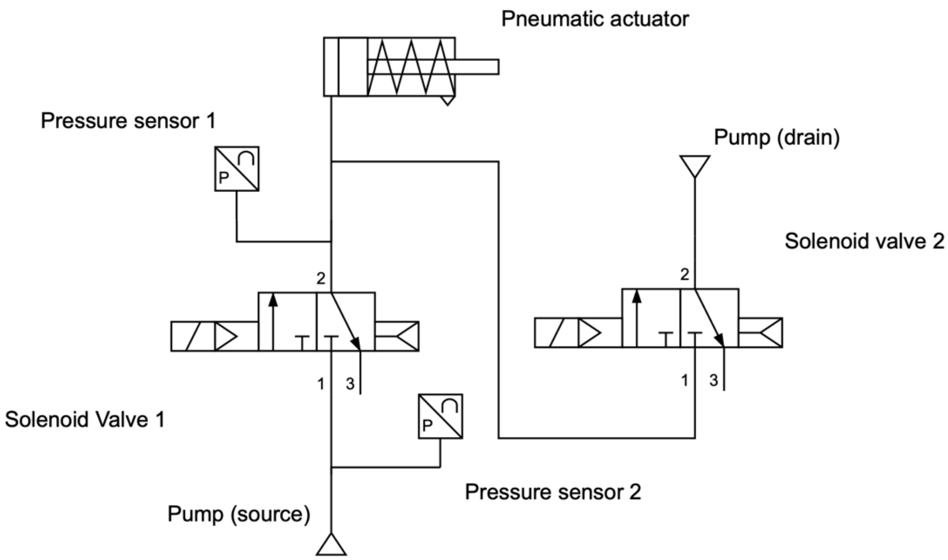
First program on myRIO:





Drawing a basic pneumatic circuit which can inflate and deflate a pouch motor:

pressure sensor connects to 4mm pneumatic tubes (the flexible ones), use a converter to convert to 6mm and then use pneumatic fittings to connect to the pouch motor



20/07/2022:

Goals:

- Create the pneumatic circuit above
- Use PID control to be able to control the pressure in the pouch motor and inflate/deflate it

Resources:

- Using motor driver

Questions:

- Are the solenoid valves bidirectional (in the product specifications they don't seem so)? If so, do I need two solenoid valves? → yes, solenoid valves are probably bidirectional & MUST: keep pouch motor inflated at a certain pressure (might need more than two, who knows)
- Motor driver won't work with myRIO controller. Why? Can you make multiple connections with the 5V pin (from pressure sensor and motor driver)? Is the motor driver getting enough voltage through it (is the potential divider circuit correct)? → Resistor gets VERY hot → solutions: 250 ohms is too low, raise the resistance, OR use potentiometer 10k ohms
- Pneumatic fitting and pneumatic tube that fit the pressure sensor?
- Pneumatic tube / fitting for out valve of the pump?
- Is pressure sensor working?

- How to scale the pressure sensor (esp with myRIO)? → Arduino gives a number (e.g.) → scale that number across voltage from 0.5 to 5V out of 1024 → scale the voltage to pressure using the chart in spec sheet
- You're supposed to separate analog ground and digital ground?
- Pneumatic tube for in valve of the pump? → brute force connect

<u>21/07/2022:</u>

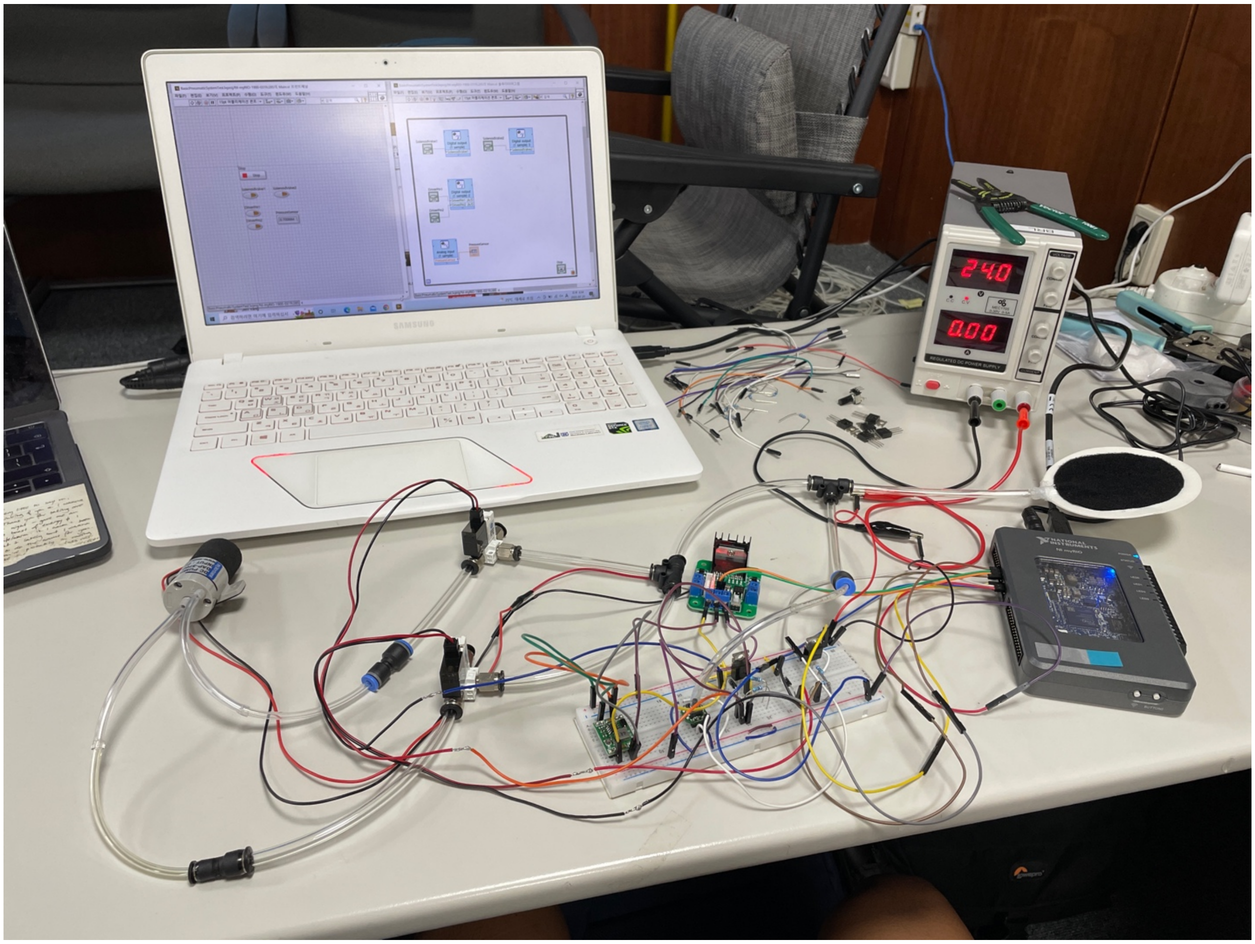
Questions:

- Motor driver only works when potentiometer output voltage is near 24V. It works at 8V and lower if we start at 24V and use the pot knob to ramp the voltage down, but otherwise we cannot use it at 6V. Why? Also the potentiometer gets very hot and the linear potentiometer was burning. → voltage regulators? Maybe on device mart?
- Potentiometer does not stick properly
- Air leaking through pressure sensor
- I need another solenoid valve, TIP 120 transistor with it
- Still having trouble with speed of the motor (cannot do PID without being able to control speed)

Resources:

- Pressure sensor specs
- Pressure sensor specs 2
- How to use voltage regulator
- DC motor control using myRIO
- Motor driver used
- How to use motor driver
- How to use motor driver
- LabVIEW plotting graphs

Electronics for the Basic Pneumatic System:

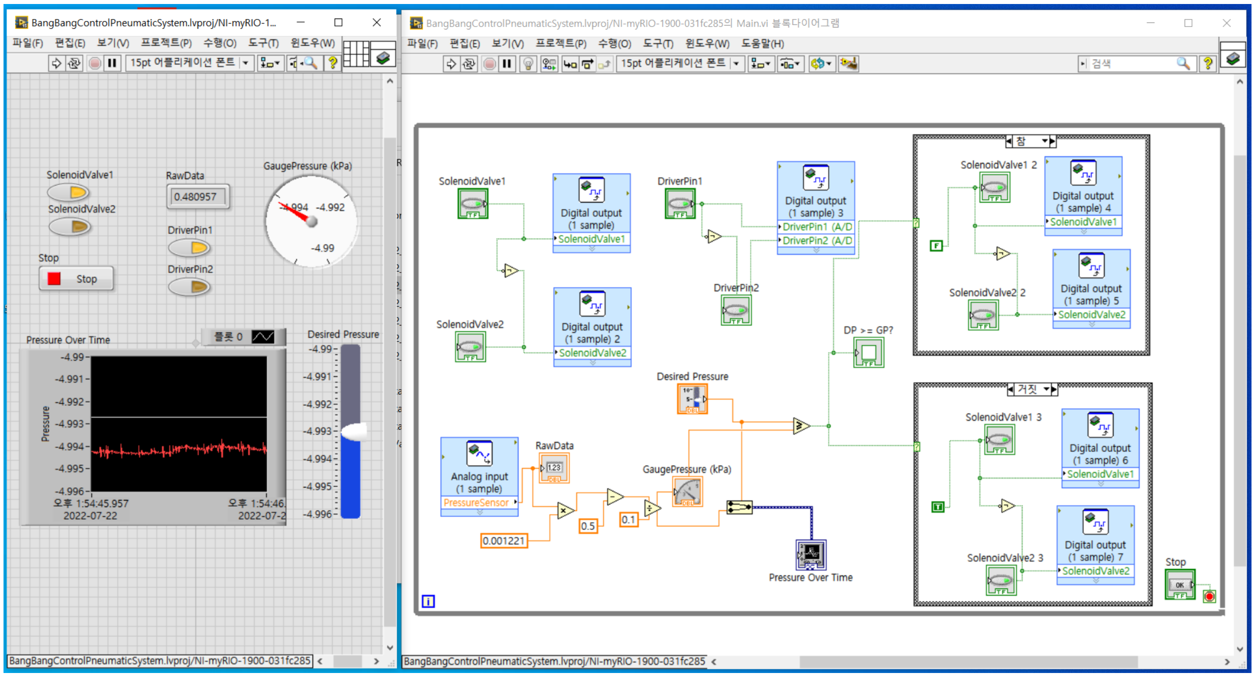


- DC supply at 24V
- Two solenoid valve circuits (digital outputs)
- Pressure sensor circuit (analog input)
- Voltage regulator circuit (to drop 24V to 6V for the pump)
- Motor driver circuit and pump (digital output)

Questions:

- Stronger pump needed?
- Air leaking through sensor and pouch motor?
- Pressure sensor readings don't make sense? Pressure doesn't change at all (not even 1 Pa change)?
- LabVIEW control and simulation module needed for control? Nop
- What next? PID control? For that I need to be able to control the speed of the pump? Another driver board?
- Change if structures into while loops?

Bang bang control:



Problems:

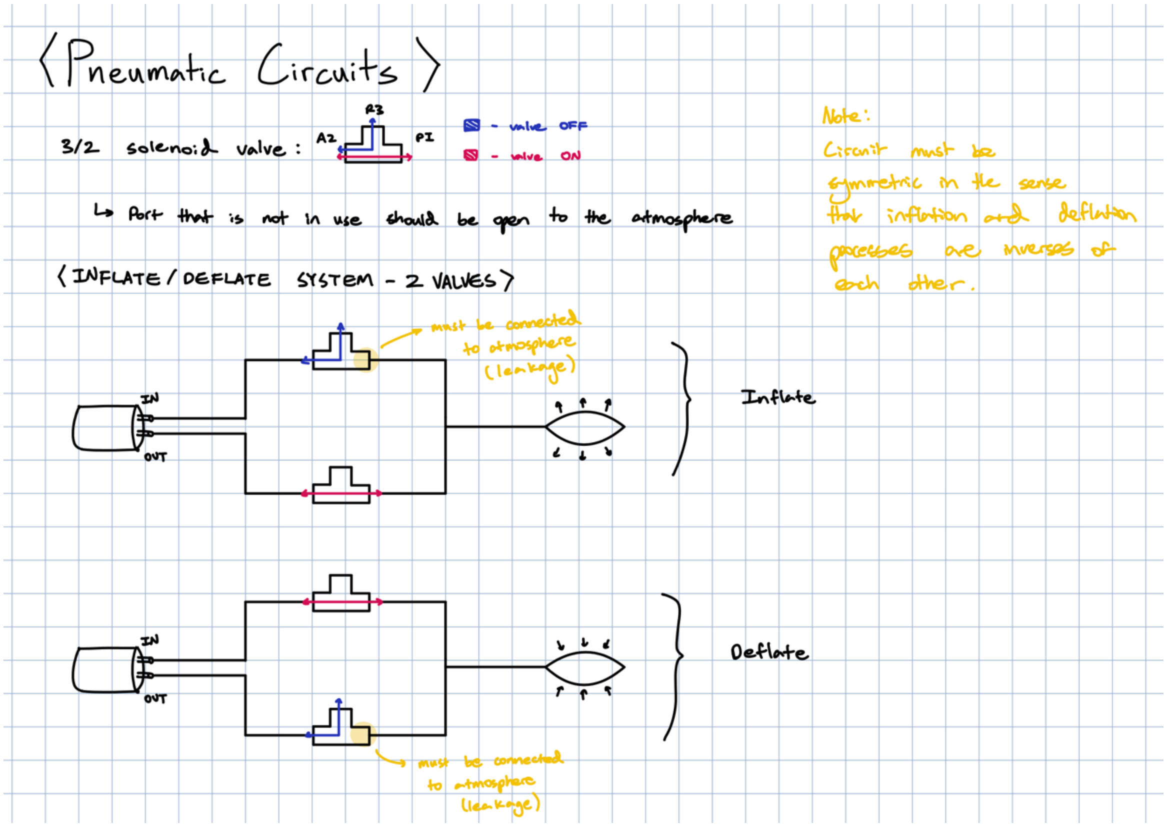
- It doesn't work
- Solved the motor driver issue
- For 3/2 valve, one valve should always be open to atm

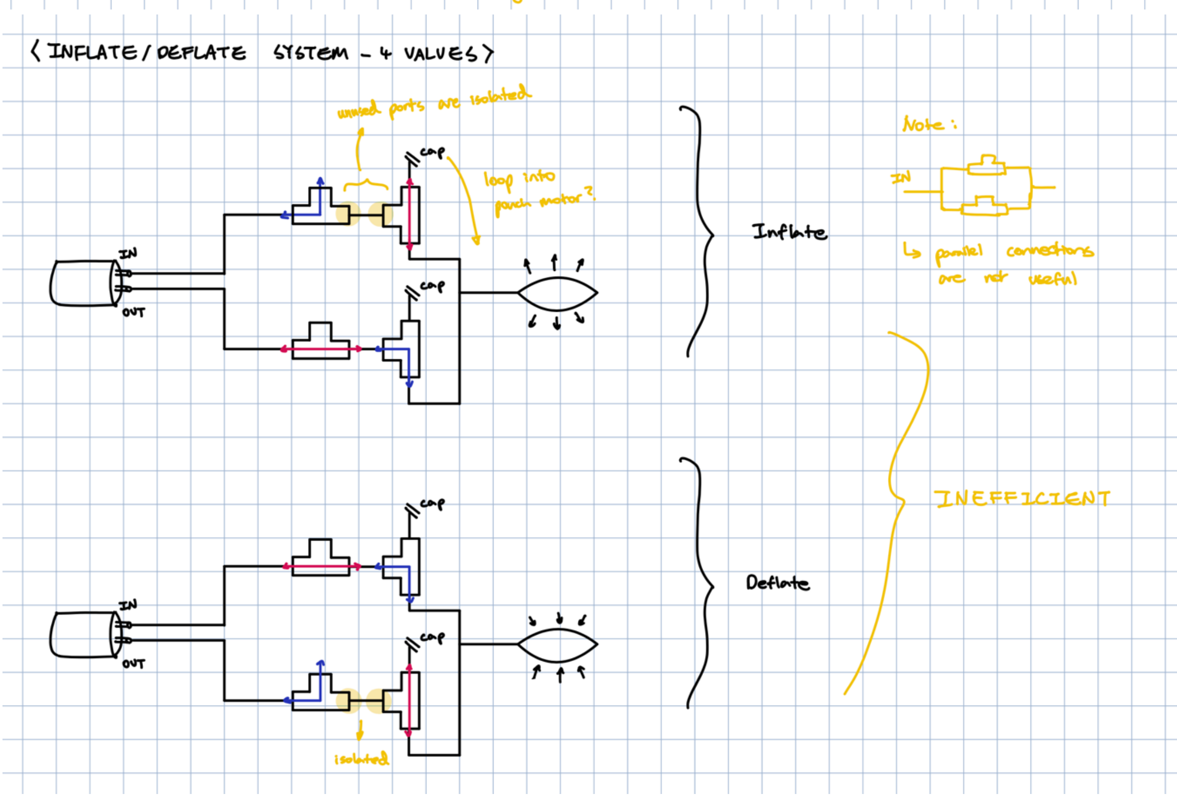
<u>To Do List:</u>

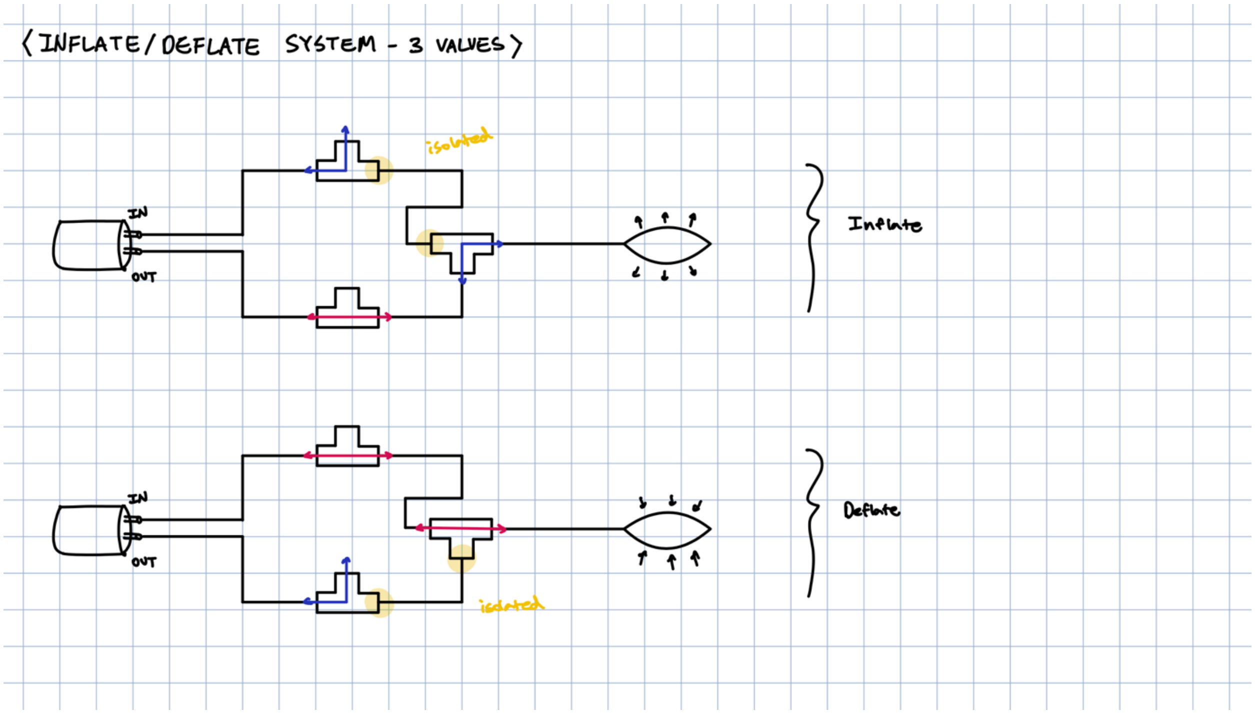
- Make new pneumatic circuit such that one valve from the 3/2 valves is always connected to atmosphere → made and shown below
- Get the pressure sensor to work properly (OUT2 to analog pin, + to 24V, - to GND) → raw data from analog input is already the voltage so just use raw data value to convert to pressure using spec sheet of the sensor
- Finish PID control LabVIEW program
- Pouch motor should be able to be inflated a lot with 100kPa pump → rewiring the pneumatic circuit fixed this
- Weekly report → done

25/07/2022:

Thinking about new pneumatic circuit:

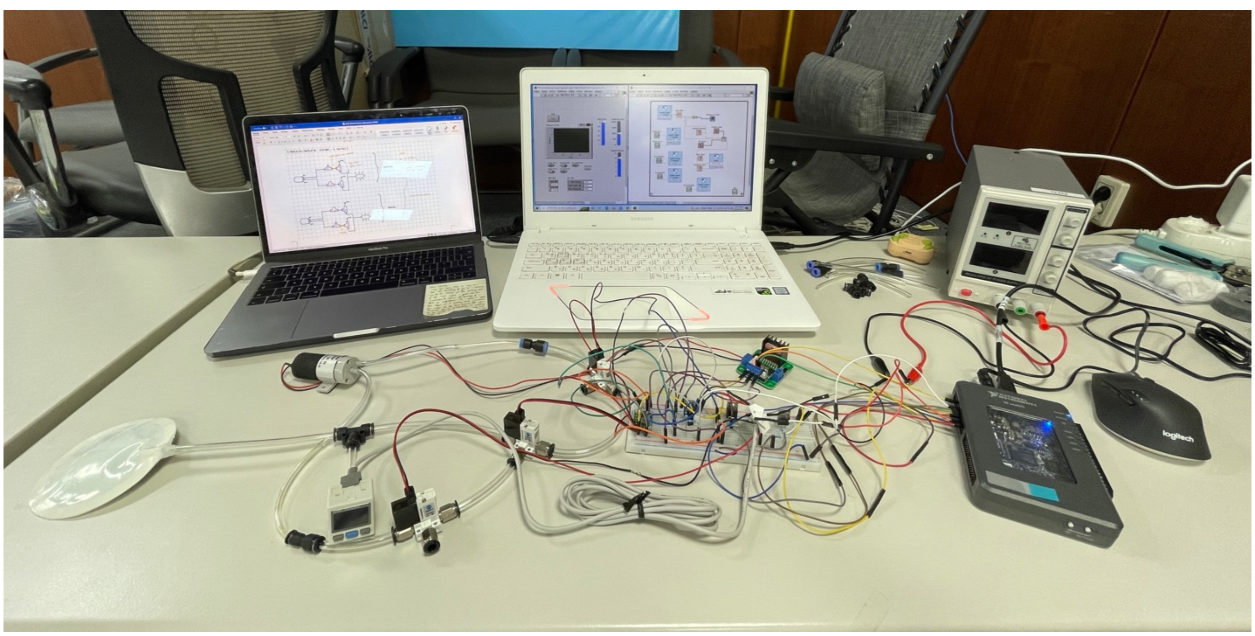






Pressure sensor range: - 100 kPa to 1.0 MPa

<u>New setup:</u>



Resources:

- [PneuSoRD]
- [PneuSoRD]

- Closed loop control LabVIEW with dc motor
- Using charts and graphs LabVIEW
- DC motor control LabVIEW
- PID control LabVIEW
- https://realpars.com/pid-tuning/
- https://www.ni.com/docs/en-US/bundle/labview/page/lvpid/pid_vi.html#:~:text=PID%20(DBL),appropriate%20for%20your%20control%20system.
- https://www.smcpneumatics.com/pdfs/ISE30.pdf
- https://www.smcworld.com/assets/manual/en-jp/files/ZISE30A.eng.pdf
- https://www.ni.com/en-us/support/documentation/supplemental/21/labview-equivalent-of-if--if-else--and-switch-statements.html

Learning how to tune PID controller:

- PID control part 1
- PID control part 2
- PID control part 3
- PID control part 4

Questions:

- What should the frequency of the PWM signal for the pump be?
- What is a shift register?
- Pressure sensor readings are weird?
- Why doesn't the waveform graph show values of pressure and desired pressure
- Why won't pressure follow desired pressure?
- Questions for the PID controller
  - Say, junction valve is blocked so pump builds higher and higher pressure, resulting in integral windup upon release → how do we fix this? → integrator anti-wind up (clamping / conditional integrator?)
  - Add filter to derivative path?

To Do List:

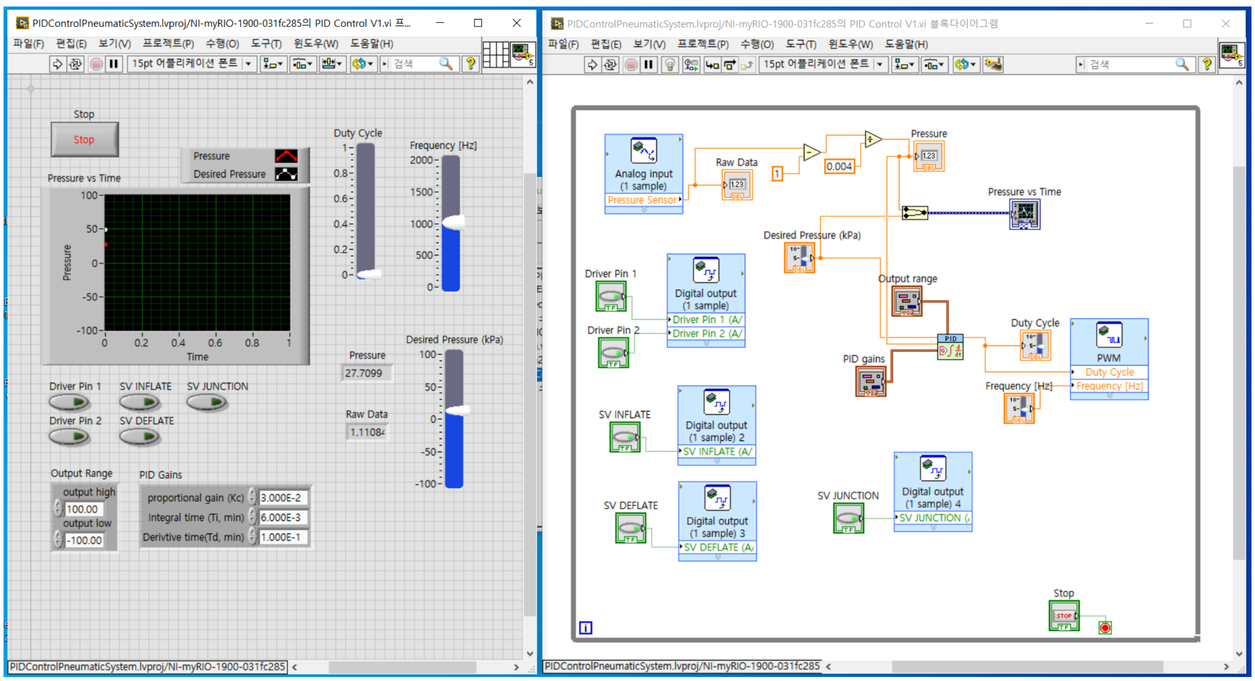
- Finish and tune the PID control system for the pneumatic system
- Fix waveform graph

Main problems:

- Waveform graph does not display pressures → write a new program → https://forums.ni.com/t5/LabVIEW/conversion-from-double-to-1d-array-of-double/td-p/2690791

- Pressure readings seem wrong → try 40kPa sensor with myRIO, try 1.0MPa sensor with the Arduino, just use the wrong values or make an arbitrary offset that roughly approximates the pressure values + CONNECT SENSOR GND TO ANALOG GROUND
- Pressure readings fluctuate a lot, messing with the control system → probably due to PID gains → AVERAGE THE VALUES
- PID controller is not tuned → overshoots, very slow to reach target pressure, fluctuates around target pressure → try a simpler program that just stops the motor when a certain pressure is reached

PID control v1 Program:



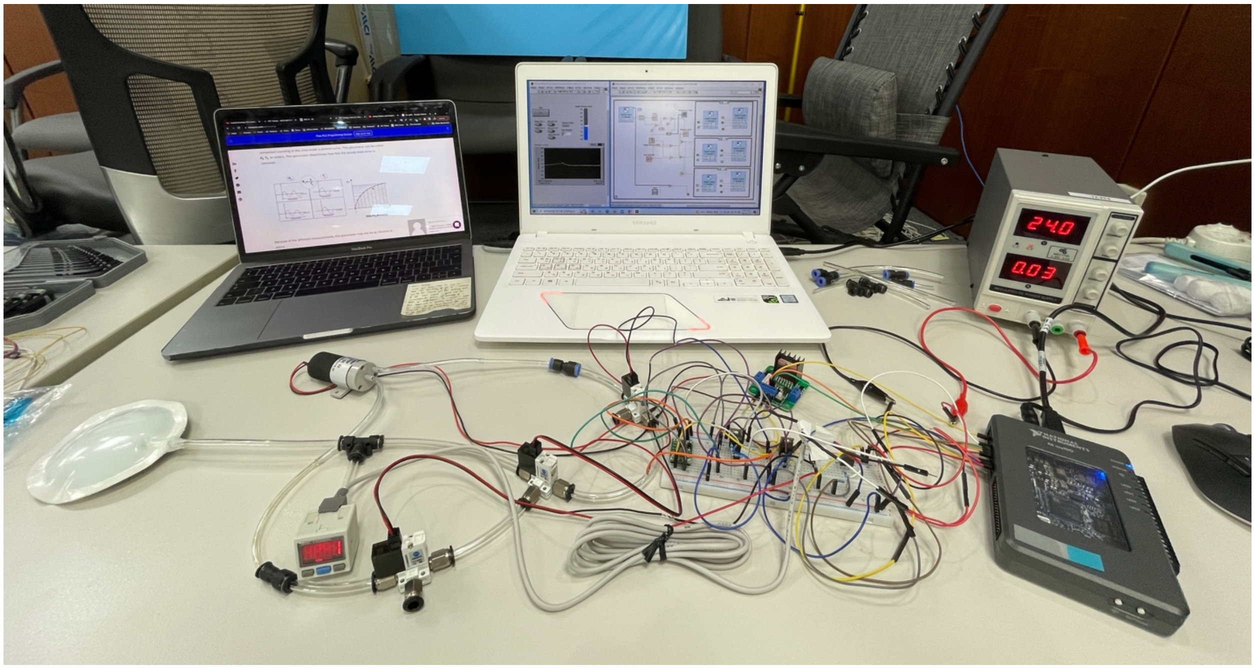
<u>26/07/2022:</u>

To Do List:

- Finish pneumatic system and control system → done → bigger pump now for faster inflation? Working with compressor / CO2 cans
  - Using higher pressures
  - Solenoid valves only support up to e.g., 100 psi → we might need a pressure regulator to use CO2
  - What are pressure regulators? How to use them? What can we buy or make?
  - Look up initial pressure of CO2 cans
  - First time using the can, the pressure will be very high → what is the range? Look it up
  - Use stepper motor to control automatic inflator → design / replicate it (probably not to small because pressures are high so expect high stresses, not too thin)

- Read about biomechanics of walking?
- Learn Autodesk Eagle?

Pneumatic System and Control System V1 Summary:



The previous pneumatic system and control system had numerous problems. Namely:

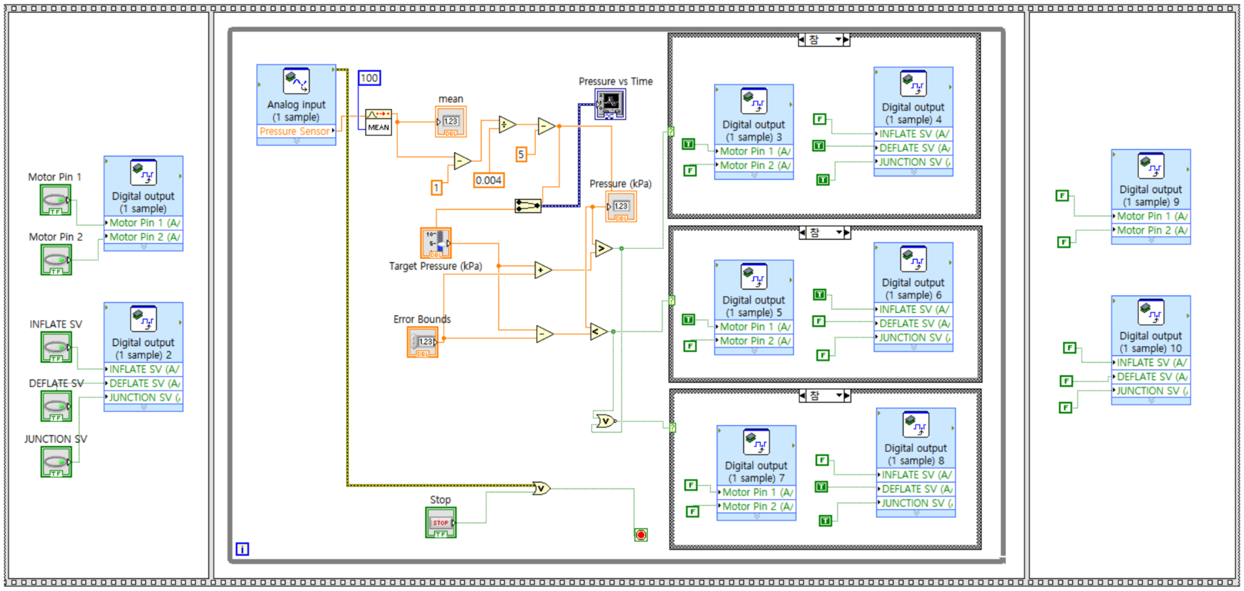
1. The measured pressure from pressure sensor gave values which fluctuated too drastically, preventing the control system to work smoothly.
2. The pressure sensor gave incorrect values.
3. The PID controller could not be tuned and was not working properly.

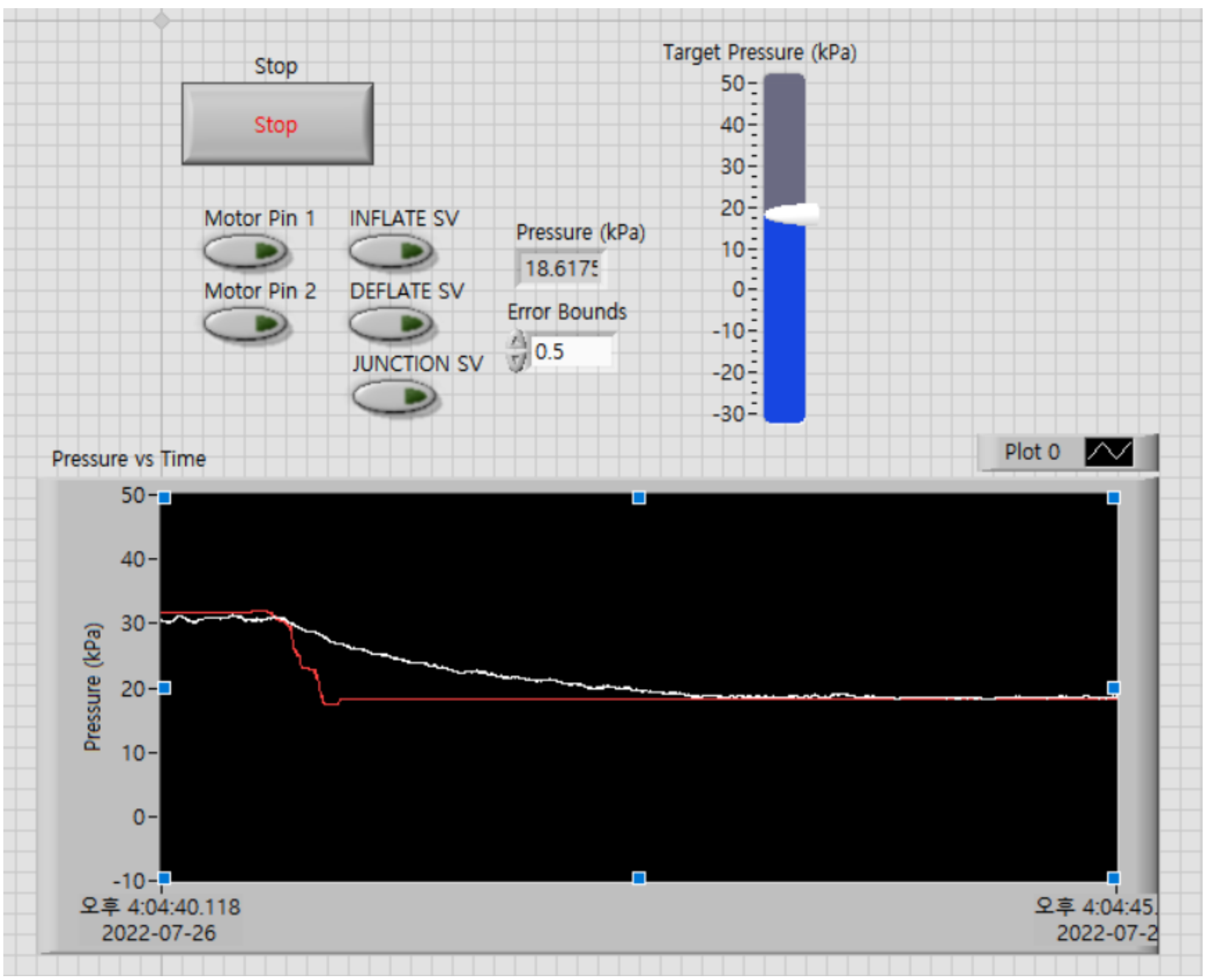
A PID control system was an over-engineered solution for a system that is as simple as the pneumatic circuit above. The flow rate of the pump is low enough such that there is no overshoot and also no need to reduce the flow rate of the pump as the vessel pressure approaches the target pressure. Therefore, a control system similar to bang bang control will be used, i.e. the pump will inflate when the pressure is a certain value below the target pressure, deflate when a certain value above the target pressure or turn off when pressure is within this “error bound” region.

It was observed that pressure can rise and drop to a target pressure and oscillate near the target pressure as desired. The drawback to this control system is that depending on how small the “error bound” value is the frequency with which solenoid valves turn on and off are very high. This may be inefficient or be harmful to the valves’ healthy operation.

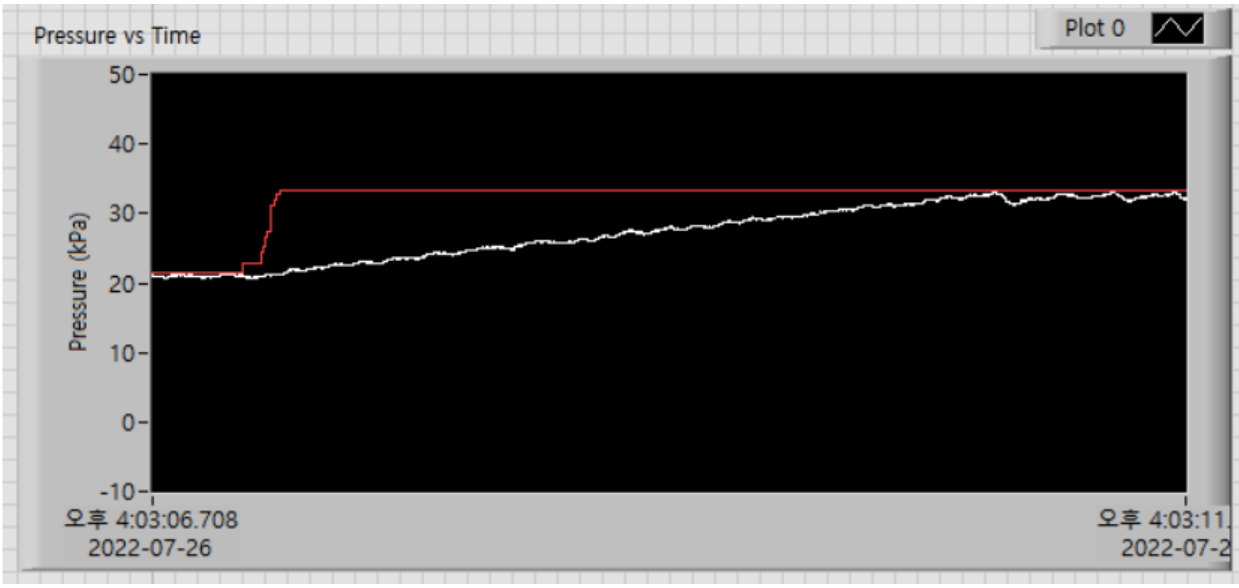
The response time for inflation seemed to be slower than that of deflation.

LabVIEW program:

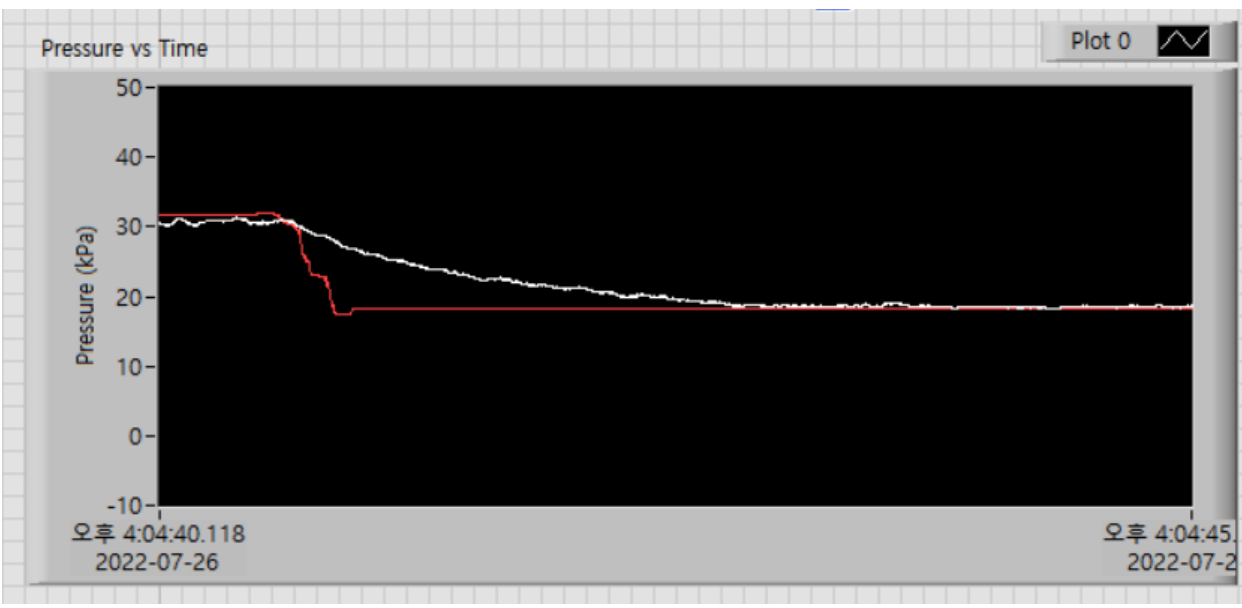




Inflation:



Deflation:



Solenoid valve specifications:

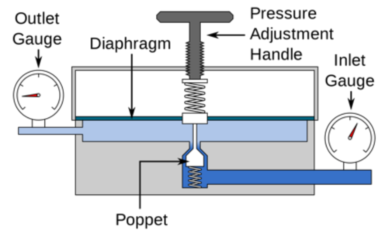
- Pressure range: -100 kPa to 0.7 MPa (equivalent to 101.526 psi)
- Solenoid valve will break if pressure from CO_2 canisters is not regulated properly

CO2 cartridges:

- Pressure inside CO2 canister is around 850 to 900 psi
- https://www.quora.com/What-is-the-maximum-pressure-inside-a-CO2-cartridge

Pressure regulators:

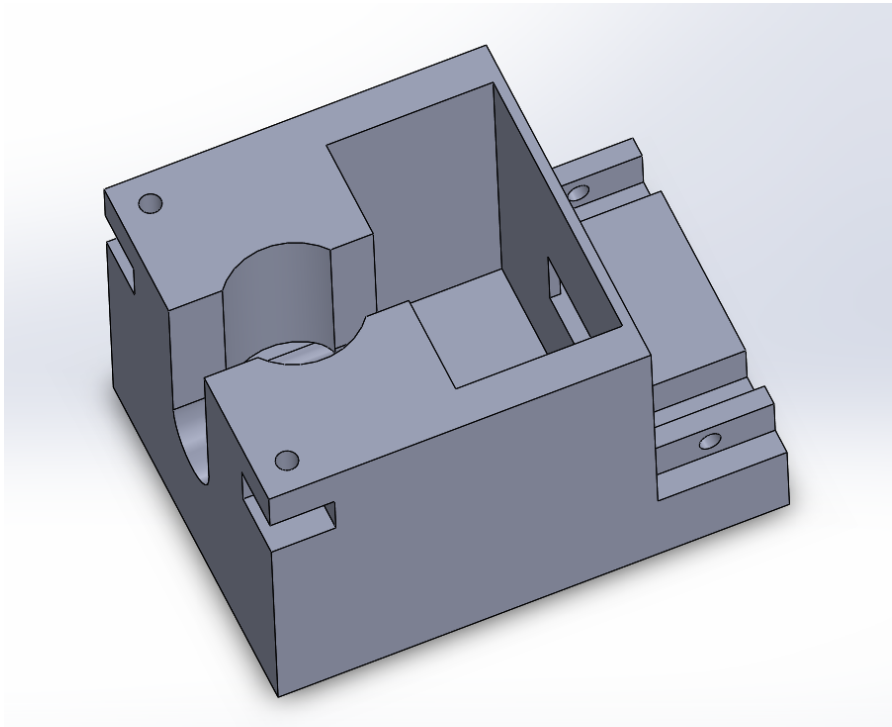
- A pressure regulator is a device which controls the pressure of liquids or gases (medium) by reducing a high input pressure to a controlled lower output pressure. They also work to maintain a constant output pressure even when there are fluctuations in the inlet pressure.

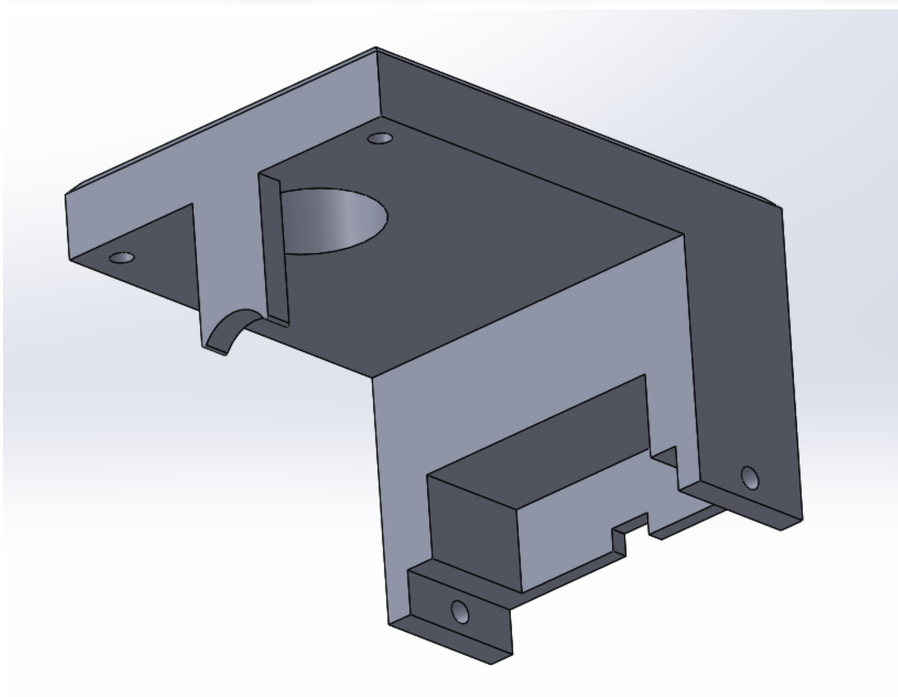


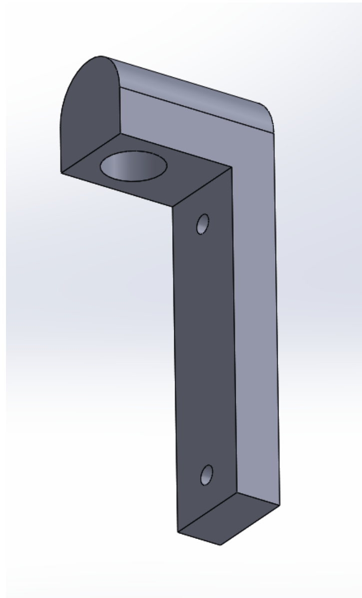
- https://en.wikipedia.org/wiki/Pressure_regulator
- https://smartstore.naver.com/jshose/products/244626217?NaPm=ct%3Dl61vpox4%7Cci%3Dec542ade9fc86c2d1104df1fa70be5d955c9eb0f%7Ctr%3Dslsl%7Csn%3D218846%7Chk%3D9ec526fa367fccedf2dd404be37a34acc8e4ab04
    - Maximum pressure (1.0 MPa) = 145 psi
    - Regulated pressure (0.05 MPa to 0.85 MPa)
- http://sym21.com/products/product.php?ptype=view&prdcode=2204190001&page=1&catcode=19000000
    - Max pressure (1.0 MPa)
- https://www.navimro.com/p/items/K21213501/?access_google_shopping&utm_source=google&utm_medium=shopping&gclid=Cj0KCQjwof6WBhD4ARIsAOi65ajy3X69b3pFSXkNEpQ8hXloSeDAWWkROP0GeejHVdWJhxZjXDBz0MAaAvU3EALw_wcB
    - Max inlet pressure (3000psig)
    - Regulated pressure (80 to 125 psig)
- http://www.drastarkorea.co.kr/?act=shop.goods_view&GS=85&GC=GD00
    - Max pressure (6000 psi)
    - Regulated outlet pressure (25 psi to 500 psi)
- http://www.gasplus.com/product/product.php?ptype=view&prdcode=1608220129&catcode=10111012000000&page=1&catcode=10111012000000&searchopt=&searchkey=
    - Max inlet pressure (20 MPa = 2900 psi)
    - Max outlet pressure (1 MPa = 145 psi)
- https://smartstore.naver.com/wonners/products/5370732713?NaPm=ct%3Dl61wgxmo%7Cci%3D0z80003CCrXwXU0n%2Df3h%7Ctr%3Dpla%7Chk%3D0aa11e79e188d59fae25297310f2601164594c5d
- https://mfckorea.co.kr/bbs/board.php?bo_table=product08&wr_id=3
- More pressure regulators
- https://mfckorea.co.kr/bbs/board.php?bo_table=product08&wr_id=3
- Check whether the threads fit CO2 cartridges we have

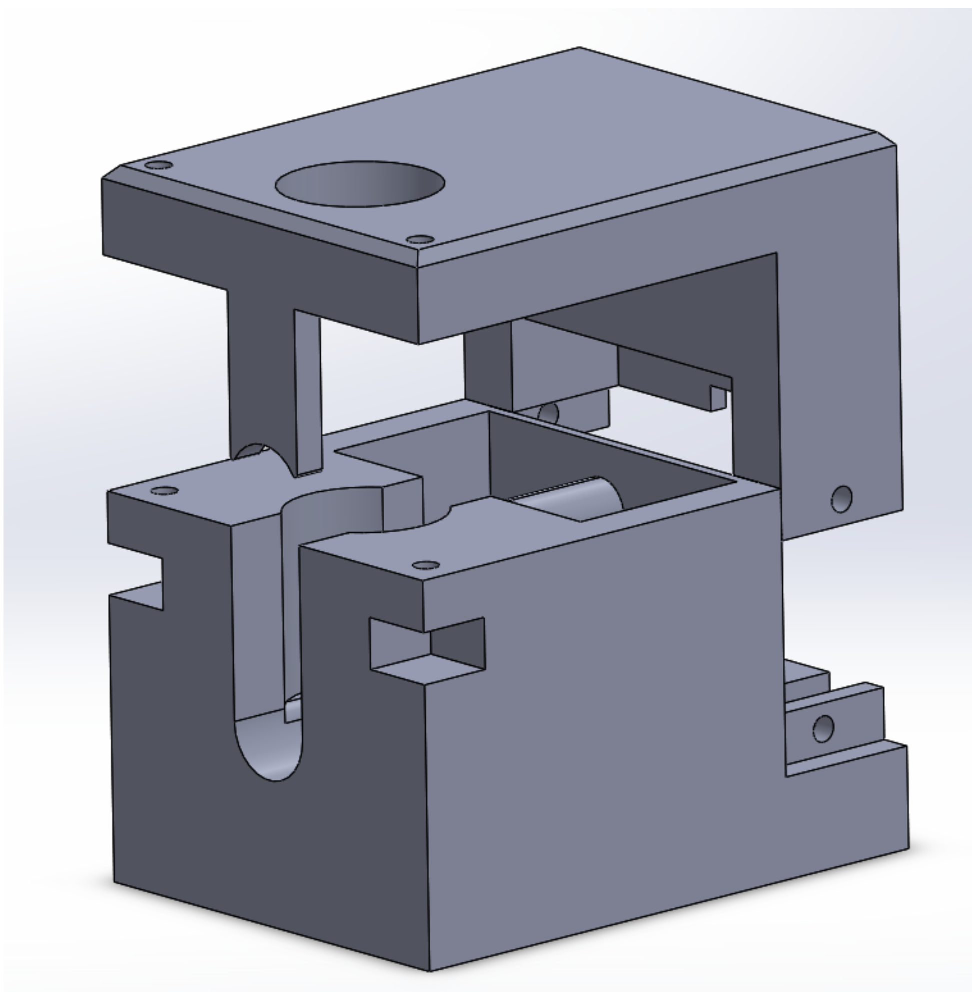
Designing an Electric Inflator:

- Device which uses a servo motor to turn the knob of the inflator which will then control the release of air from CO2 cartridges









To Do:

- 3D print parts for the inflator, assemble the inflator and control it using the servo motor
- Think about how the circuit can be made more modular → currently, if we had four pouch motors, we would need four times the circuit size (?), how can we expand the circuit
- Finish research about pressure regulators and what to buy, making sure thread sizes fit with the CO2 cannisters

Pressure Regulators:

- CO2 cartridges come in 3/8" thread sizes (3/8" to ¼" adaptors??)
- Most suitable for us: Single Stage Low Pressure NPT type (1/4") 072X-0000-1S
  - Price: 198,000 won
  - Depending on the model inlet pressure range can be chosen to be 3600 psig or 600 psig
  - Depending on the model, outlet pressure ranges can be chosen from: 25, 50, 100, 250, 500 psig
  - Depending on the model flow capacity (valve flow coefficient): CV = 0.06 or 0.2
  - Depending on the model, you can choose: 2, 3, 4, 5 or 6 ports
  - Inlet and outlet gauges are optional but will add extra cost (22,000 won or more)
  - 1kg weight
- For adaptors: Pneumatic tube to ¼ thread adaptors (inflator → thread to 4mm tube → pneumatic tube → 4 mm tube to ¼" thread → pressure regulator → ¼" thread to 4 mm tube → pneumatic tube)

To Do List:

- Design of a dynamic cushion (for passive propulsion)
  - Pouch motor / cushion design where a small volume of air causes large angular rotation (on SolidWorks)
  - Current motor is made with TPU, manufacturing is a bit dubious though
  - Look at PneuNets for inspiration maybe?
  - Read a little on the biomechanics of running (literature can be confusing and contradictory)?
- Experiments with the pneumatic system + page-style documentation
  - Pneumatic systems → things we can control right now are pressure and flow rate → CV = aperture size → we have other solenoid valves with different aperture sizes and different tube sizes → what effect does this have on inflation/deflation processes? + what about controlling number of outlets? What effect does that have on inflation/deflation processes → study effect of tube size and number of outlets on how fast we can deflate the pouch → for this to be possible we need to be able to measure pressure very well → to make the pressure display sensor work, maybe put a 1k ohm resistor between pressure sensor and myRIO which is the load? Refer to circuit diagram in spec sheet)

- Pressure sensors in series → allows us to measure pressure differences → this allows us to calculate flow rate (for pressure measurements, use barometer chips and along with that use Teflon tape, *3d print → airtight! (methods other than FDM → objet printer SLS?)* or adhesive?
- Try the pneumatic system with CO_2 cartridges, inflator and pressure regulator
  - How to connect the pneumatic system such that we can deflate the pouch motor using the CO_2 cartridges → or are we just deflating without a pressure sink?
  - Try whether the control system works
  - Design a new control system that can set pressures in a pouch motor when dealing with higher pressures and flow rates.
- Making the circuit / system more modular
  - Improve UI: Make LabView UI more intuitive (e.g. if you press an inflate or deflate button → it presses all the buttons involved with inflating/deflating, i.e. no need to manually)
    - Create secondary buttons for all valves such that if the pneumatic circuit is changed, you can configure these secondary buttons in a way such that the inflate and deflate buttons still work as they should
  - Solder boards (make solenoid valve circuits on each of these boards)
  - Learn Eagle PCB for later so that an actual PCB can be made
- Electric inflator
  - Obtain 3D printed parts and assemble it
  - Program servo motor for the inflator and test it using the CO_2 cartridges
- Send info on pressure regulators, adaptors and pneumatic fittings
  - Find out what to connect on the non-thread part of the pneumatic fittings and send links

Pressure Regulator Summary:

- Pressure regulator to purchase: Single Stage Low Pressure NPT type (1/4")
  http://www.drastarkorea.co.kr/?act=shop.goods_view&GS=85&GC=GD00
- Specifications we probably need / should choose:
  - Inlet pressure: 3600 psig
  - Outlet pressure: 100 psig, 200 psig or 500 psig (for reference, solenoid valves can handle a maximum of 101 psig)
  - Ports: 2 ports
  - Add pressure gauges with extra costs if we do not have pressure gauges
- Pneumatic fittings
  - https://smartstore.naver.com/ftsquare/products/5290353147?n_media=11068&n_query=%EA%B3%B5%EC%95%95%ED%94%BC%ED%8C%85&n_rank=2&n_ad_group=grp-a001-02-000000019246997&n_ad=nad-a001-02-000000119156376&n_campaign_type=2&n_mall_id=ncp_1o38cb_01&n_mall_pid=5290

353147&n_ad_group_type=2&NaPm=ct%3Dl64mbhcw%7Cci%3D0Bi0003y8Y9weGvEjL2l%7Ctr%3Dpla%7Chk%3D0cbdaa799c9c00aae25772483e701172cf7f8f1a#
    - We need a fitting for the ¼" threads of the pressure regulator, so "TM type" with ¼ threads (TSP 02TM)
    - If we want to fit CO2 cans directly to pressure regulator, we may also need pneumatic tube to 3/8" thread connector TP type (TSP 03TP)

Pneumatic fittings summary:

- Two touch fitting BSBM CC
    - http://www.cdcpneumatics.com/open_content/product/view.php?pm_no=133
    - A fitting which converts from male threads to pneumatic tube
    - Max pressure is determined by the tube used

- TSP-TS
    - http://www.cdcpneumatics.com/open_content/product/view.php?pm_no=199
    - A socket which converts from female thread to plug
    - Max pressure is 725 psi

- TSP-TP
    - http://www.cdcpneumatics.com/open_content/product/view.php?pm_no=200
    - A plug which converts from plug to female thread
    - Max pressure is 725 psi

- TSPM
    - http://www.cdcpneumatics.com/open_content/product/view.php?pm_no=241
    - A plug which converts from plug to male thread
    - Max pressure is 725 psi

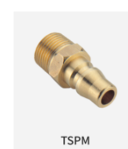
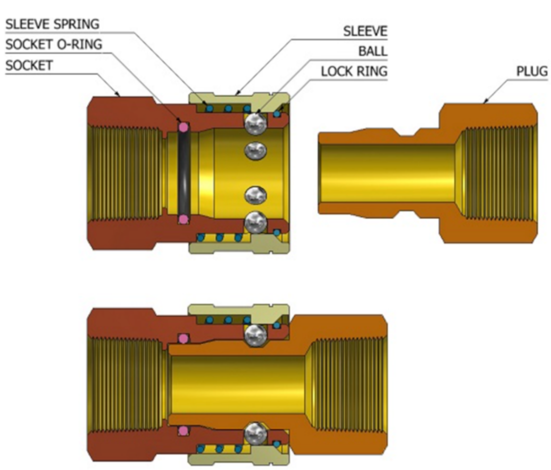


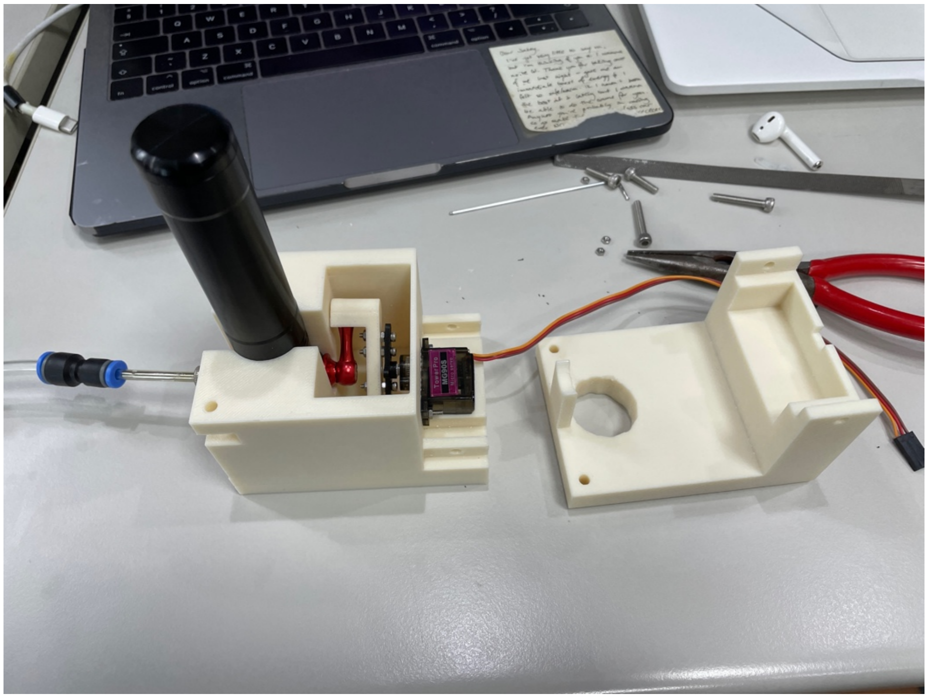
Diagram for TSP-TS/TSP-TP/TSPM sockets and plugs:



Questions:

- Why does the dynamic cushion not inflate/work? → Dynamic cushion works with the CO_2 cans (higher flow rate) but does not work with air pump → any airflow from the CO_2 cans causes the pneumatic tube used at the opening to freeze (due to decompression) → how to handle the freezing / is it ok? → freezing is ok / nothing we can really do about it
- How to connect pneumatic circuit for deflation (using a pressure source such as a CO_2 cartridge)? → use pneumatic vacuum generator (air powered vacuum) or vacuum pump or just let the pouch motor deflate naturally to the atmosphere → look for 3 or 4 vacuum generators (or look for vacuum generators with multiple ports but just buying multiple might be preferable)
    - KSV-10HS
    - ACV-15HS
- Valve diameter is different but tube diameter is the same? → make a table of sorts (pair 4 mm diameter tubes with smaller valve sizes and larger 6 mm tubes with larger valve sizes)
- Stronger servo motor is needed for the electric inflator → purchase a new servo motor (use force gauges to determine how strong they need to be) OR gears (probably not) OR look at other types of CO_2 inflators → either look for them in the lab or purchase
- How to design pressure sensor housing (for 3D printing) such that it is airtight? → we're going to use SLS material (?) which is slightly less rigid than the FDM printer so that will help, just make the housing as tight as possible + we can use a combination of methods (put adhesives, rubber ring, zip ties around the casing) + incorporate a tube holder in your design so that the pneumatic tube does not slack and create an opening
- Need more adaptors and bigger ones for the bigger solenoid valves → find them at the side of her desk
- Making pouch motor with 2, 3, 4, … ports? → when can I do this, might need a little refresher → do this at a later stage (do the valve size experiment first)
- Pneumatic circuit plan (→ plan differs pneumatic circuit differs depending on whether we use vacuum generator or not) → plan is ok (it's a bit complicated) → use 2,3,4 vacuum generators
- Proportional valve → this can indeed be used to control the flow rate which will determine how fast the pouch motor will be inflated and deflated but for now just think about controlling the pressure inside the pouch motor
- Do the valve size experiment first and while doing that order and make things needed for the multiple port experiment (vacuum generators and pouch motors)
- For the experiments, use the pump instead of the CO_2 cans as the cans change pressure and flow rate as they are used (try using barometer chips to measure pressure and flow rate)
- For the barometer chips, calibrate pressure sensors manually

Electric Inflator Assembly:

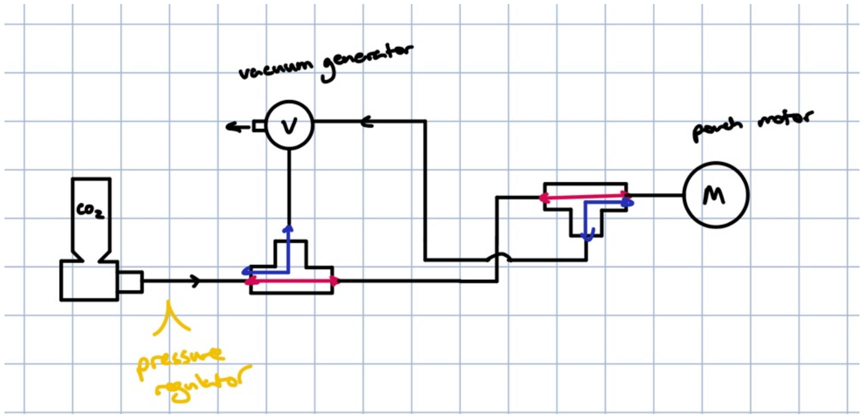


Problems:

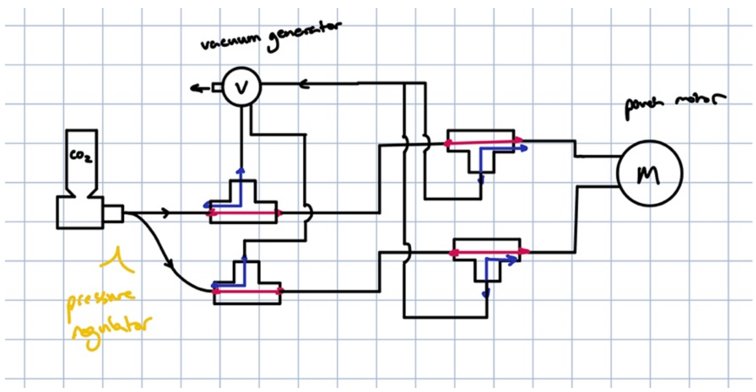
- Servo motor does not generate enough torque to turn the knob on the inflator

Plan for pneumatic circuit when varying number of ports of the pouch motor:

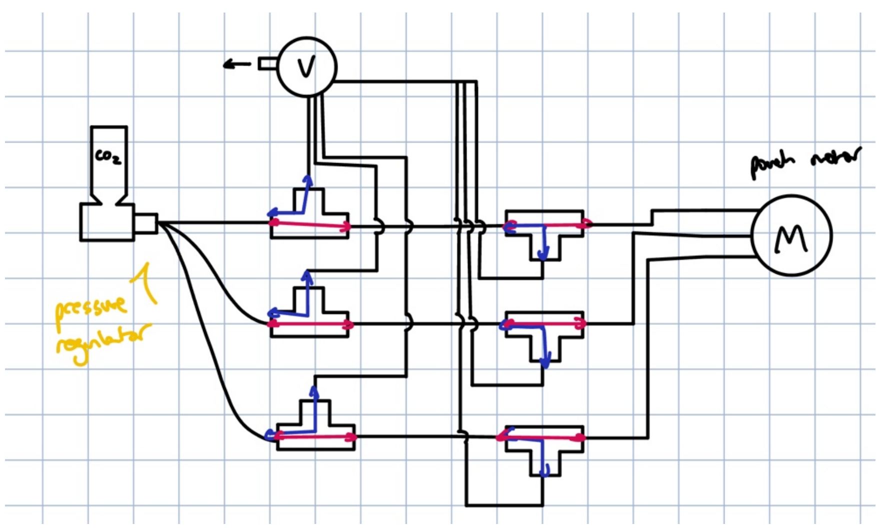
One port:



Two ports:



Three ports:



Note: For a n-port pouch motor, we need 2n solenoid valves

To-Do List:

1. Use force gauge to measure the torque needed for the CO_2 inflator
2. Find and buy the appropriate servo motor
3. Search for vacuum generator and buy the appropriate quantity
4. Draw pneumatic circuits for valve diameter experiments and multiple ports experiments
5. Design 3D printed housing for pressure sensor
6. Print the housing OR for now use the display sensor
7. Make sure pressure sensor mount fits and works
8. Calibrate the pressure sensor, figure how to properly wire it and get proper raw data
9. Buy ¼" thread to pneumatic tube converter
10. Write weekly report
11. Do valve diameter experiment
    a. Look for adaptors on HyeJu's desk
    b. Wire all the electronics
    c. Wire the pneumatic circuit (using the pump)
    d. Create a new LabVIEW UI which makes it easier to switch between pneumatic circuits (inflate deflate buttons and buttons to change configurations)
    e. Make a table for the valve diameters and tube diameter combinations that will be tested
    f. Perform the experiment, collect data and write (figure how to calculate flow rate from pressure differentials)
12. Make pouch motors with multiple ports
13. Do the multiple port experiment
    a. Wire all the electronics
    b. Rewire the pneumatic circuit (using vacuum generators)
    c. Make a table with valve diameter combinations used
    d. Perform the experiment, collect data and write
14. When servo motor arrives, redesign the electric inflator (gears? More torque? Better mount? Less mass?)
15. Test and program for the electric inflator first on Arduino then on LabVIEW
16. When pressure regulator arrives and electric inflator is ready, rewire the pneumatic system and design a new control system for the setup
17. Design a new dynamic cushion with causes larger angular movement from a relatively small volume of air
18. Learn Autodesk Eagle and think about PCB design / making the circuit more modular (on solder boards maybe?)
19. Read journals and other resources to understand modes of injury, landing impact and muscle vibrations when running
20. Start making a list of materials and components used for the project (table style)

Force gauge experiment on inflator:

- Peak force needed = 54.1N
- Max moment arm = 23 mm
- Maximum torque needed = 1.244 Nm
- Likely force needed = 30N
- Likely moment arm = 15 mm
- Likely torque needed = 0.45 Nm

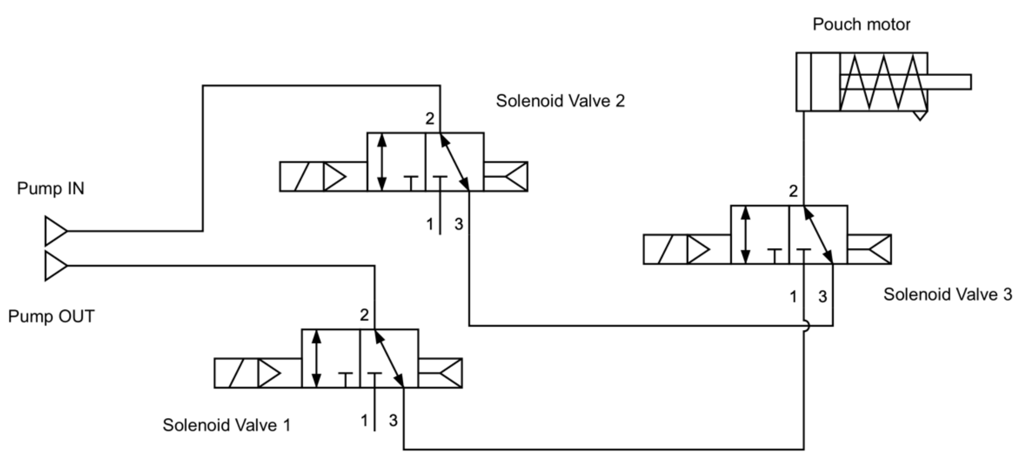
New servo motor:

- DS3235 – around 3 Nm of torque → 180 degrees of travel
- DS3120MG – around 2Nm of torque → only 90 degrees of travel
- MG996R – around 1 Nm of torque → 180 degrees of travel → buy this one

Vacuum generators:

- Vacuum generator KSV-10HS → 1/8” inlet ports
- Vacuum generator KSV-15HS → ¼” inlet ports
- Vacuum generator SMC ZH05B

Valve diameter experiment pneumatic circuit:

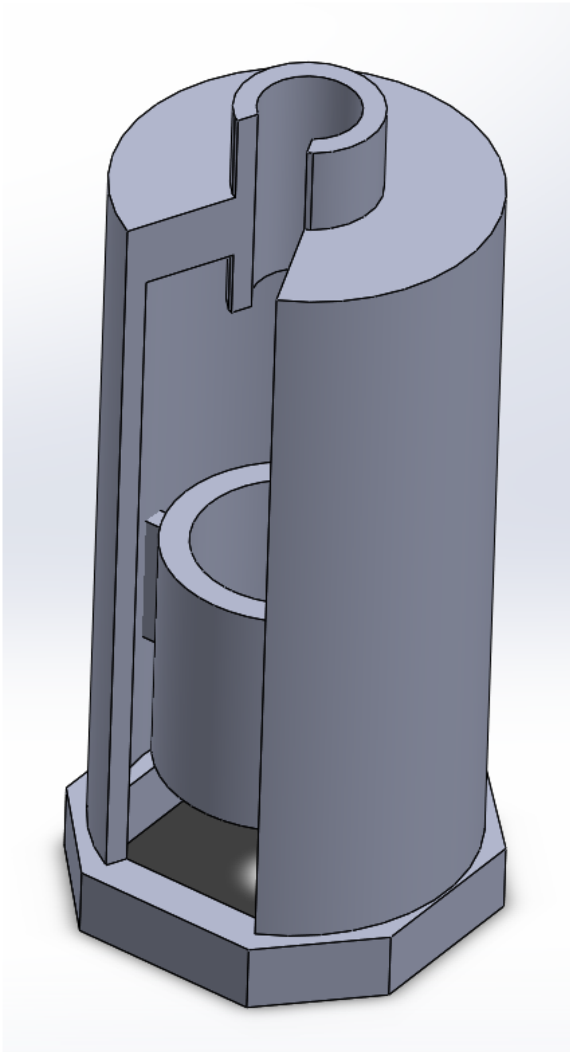


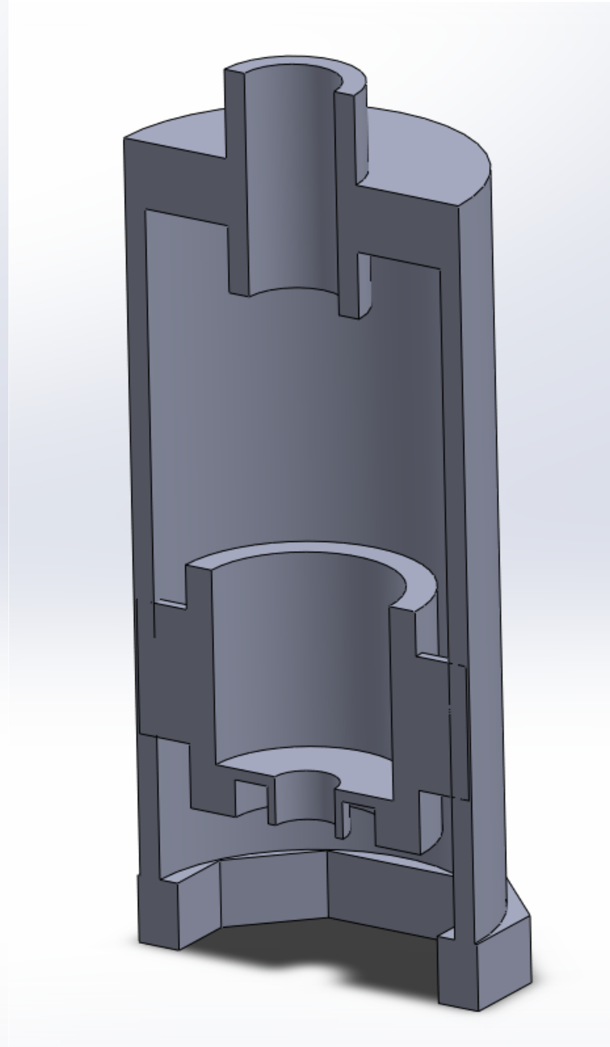
<u>More pneumatic fittings:</u>

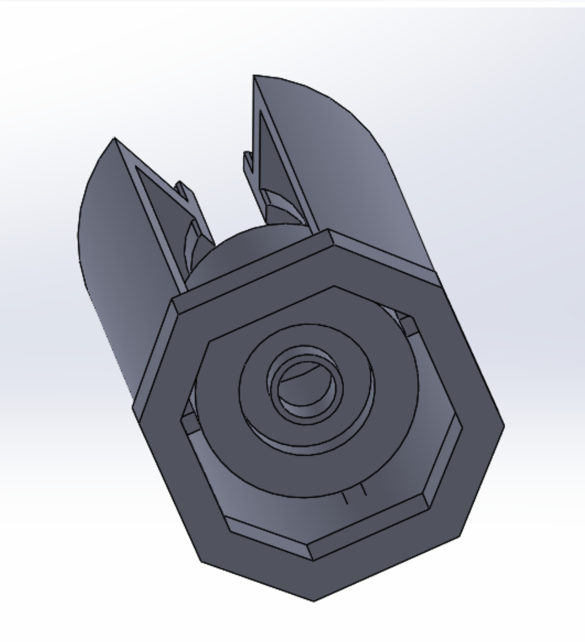
- http://www.cdcpneumatics.com/open_content/product/view.php?pm_no=133 (tube’s pressure)
- http://www.cdcpneumatics.com/open_content/product/view.php?pm_no=1 (0 to 150 psi)
-

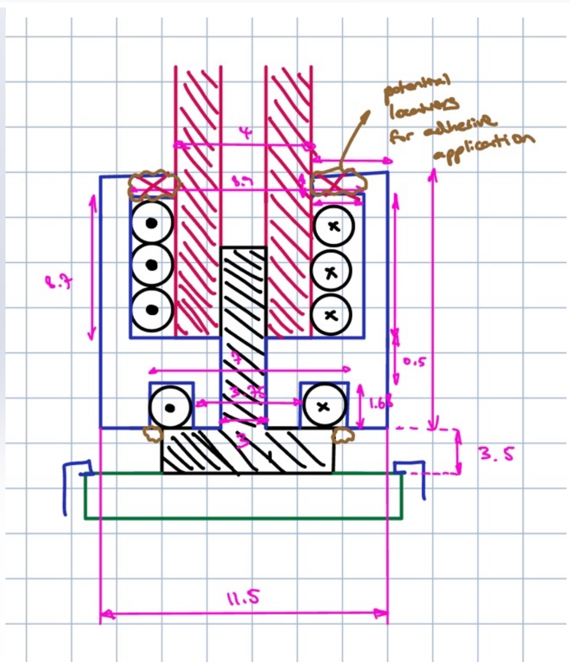
Pressure Sensor Mount Design:

- Pressure sensor tube is 3 mm in diameter



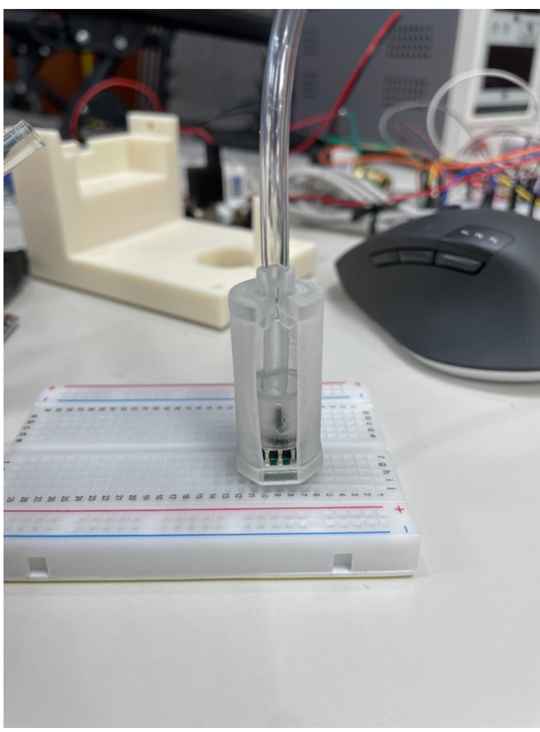





potential
locations
for adhesive
application
4
8.7
8.7
3.75
0.5
3.5
11.5

Pressure Sensor Mount 3D print results:

- There were numerous problems with the 3D printed mounts
- Problems with both methods of printing
    - Very difficult / impossible to assemble, especially the three rubber rings designed to go around the pneumatic tube
    - The angle of opening for the sensor mount is too small, making assembly difficult as well as making it too hard to insert the pneumatic tube
- Problems with SLS printed sensor mount
    - Residue was very difficult to clean up on the inside of the sensor mount
    - The extrusion between the pressure sensor and lower rubber ring broke easily
    - The lower rubber ring fit but it was a bit tight, so it kept slipping out
    - No way to ensure that the pressure sensor is vertically tight with the mount such that no air leaks through the lower rubber ring
- Problems with FDM printed sensor mount
    - The extrusion between the pressure sensor and the lower rubber ring did not print
    - The hole for the pressure sensor was way too small
    - The lower rubber ring did not fit well
    - For one of the sensors the extrusion for the lower rubber ring did print but it was too tight so the rubber ring did not fit




Circuit rewiring:

myRIO ports

- AI0 to AI2 (3,5,7) → three barometer chips → white
- AI3 (9) → display pressure sensor → white
- AGND (6) → barometer board ground → blue
- DGND (8) → solenoid valve board ground → blue
- +5V (1) → barometer board +5V → purple
- DIO0 to DIO8 (11,13,15,17,19,21,23,25,27) → up to 9 solenoid valves → green
- DIO9/PWM1 (29) → motor driver ENA → orange
- DIO 14 and DIO 15 (32,24) → motor driver INA and INB → orange

LabVIEW setup

- Every pneumatic circuit will have a certain "valve configuration" (9-bit binary number) which represents the combination of polarities for the solenoid valve that cause circuit to inflate the pouch motor
- Solenoid valves (left to right) on the circuit board will be connected to the myRIO ports (right to left, i.e., 11 to 27)

TO DO (URGENT):

1. Calibrate pressure sensors for current pneumatic circuit using the motor as a pump (put data in Excel)
2. Reconnect circuit with compressor
3. Make new LabVIEW UI
4. Try Simple Control System for the current pneumatic system → works but there is quite some oscillation
5. Make a table with the combinations of valves that will be tested for the experiments / make a more specific experiment plan
6. Do valve diameter experiment → test at night in 306 ho
7. Do multiple ports experiment → get from hyeju on Tuesday (?) and test at night in 306 ho
8. Process experiment data and collect more data if needed
9. Develop a useful way to show data (valve diameter, number of ports, power required or weight on 3 axes?)
10. Start preparing for the presentation
11. Make hand valve flow controller
12. Create a new control system that works better (less oscillations and follows target pressure closely and more responsively), PID controller?
13. Think about and research control strategy and write a program that allows pressure to follow target control strategy (What curve should the target pressure be? What frequency?)
14. Non-leaking pouch motor design?
15. How to deal with leaking pressure sensors?

    Test control system with drop test or walking test?

Pressure sensor calibration:
https://docs.google.com/spreadsheets/d/1wc_E3_NpJjFLca-kbtxfTpixi3LcXkG4tlVXOK0Z1gs/edit?usp=sharing
Question: can the compressor supply constant pressure?

Control Strategy:

1. Reduce landing impact
2. Reduce muscle vibration

Source	Key information
"An Investigation of Improved Airbag Performance by Vent Control and Gas Injection"	

Valve diameter experiment:
Solenoid valve specifications (all valves take up to 0.7 MPa):
UV-480-4L

- P(1) → A(2) effective area 4.0 mm^2 (Cv 0.22)
- A(2) → R(3) effective area 4.0 mm^2 (Cv 0.22)

UV-280-4E

- P(1) → A(2) effective area 1.2 mm^2 (Cv 0.066)
- A(2) → R(3) effective area 1.2 mm^2 (Cv 0.066)

V290-4E

- P(1) → A(2) effective area 0.3 mm^2 (Cv 0.018)
- A(2) → R(3) effective area 0.4 mm^2 (Cv 0.024)

KV-190-4E

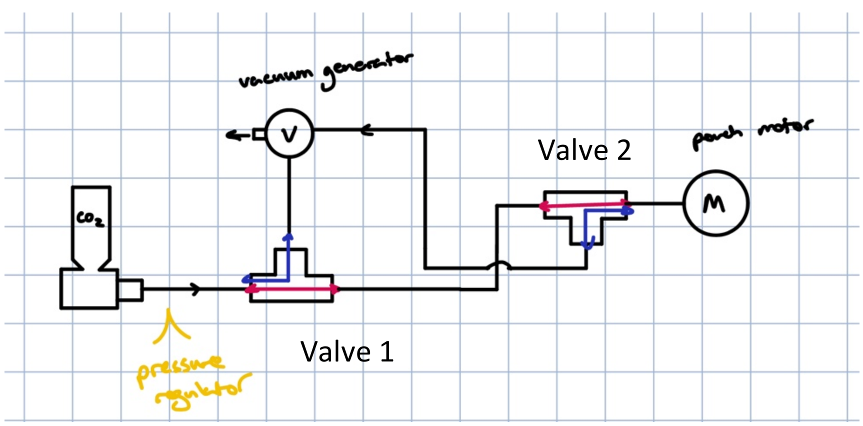
- P(1) → A(2) effective area 6.0 mm^2 (Cv 0.33)
- A(2) → R(3) effective area 6.0 mm^2 (Cv 0.33)

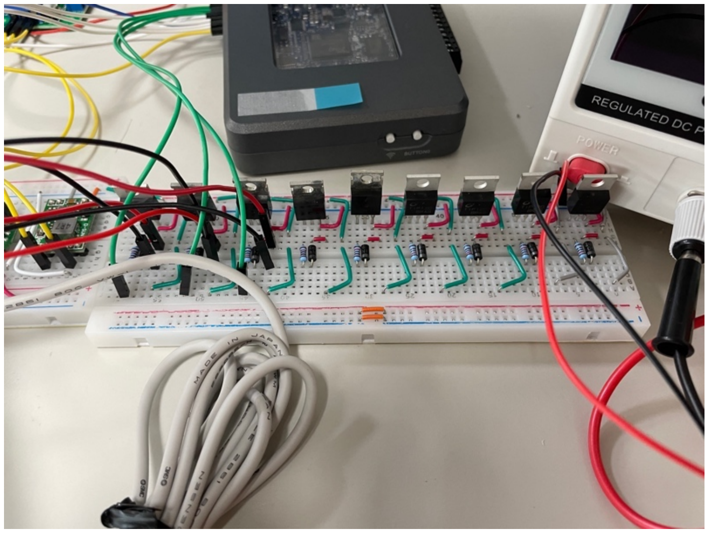
1. Increase pressure in compressor until it reaches 30 psi

2. Make sure pressure regulator is set to 1 bar
3. Make sure pressure inside pouch motor is 0 kPa
4. Run the LabVIEW program
5. Turn on the hand valve from the compressor
6. Record response time for inflation to 10 kPa
7. Repeat experiment for inflation to 20 kPa, 30 kPa, 40 kPa, 50 kPa, 60 kPa
8. Start with pouch motor pressure at 60 kPa
9. Deflate to 0 kPa and measure the response time for deflation
10. Repeat experiment for deflation from 50 kPa, 40 kPa, 30 kPa, 20 kPa, 10 kPa

<u>Questions:</u>

Keeping compressor flow rate and pressure constant?
Tube diameter and effective area both change when changing a solenoid valve?
Multiple port pouch motor?
1/16 thread converter for KV290?
Flow rate calculation? Darcy equation? Proportional to square root of head loss? Pressure difference from where to where? What if difference is not large enough or not large enough to be detected by pressure sensors and their calibrations? Should flow rate just be measured as time taken to inflate to desired pressure?
Pressure sensors both change to 100kPa?
Cannot find specs for UV480?
Change LabVIEW program so that we can measure response time?
Can I take computer home, will help me make presentation

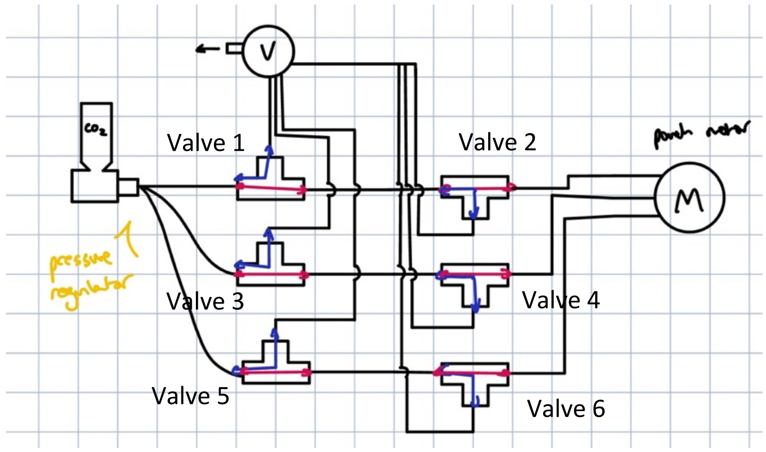




Valves will be connected from left to right on the circuit board. So, the Valve 1 will be connected to the leftmost solenoid valve circuit. The “code” will also be from left to right. So, 100000000 means that to inflate Valve 1 must be turned on and Valves 2 to 9 must be turned off.

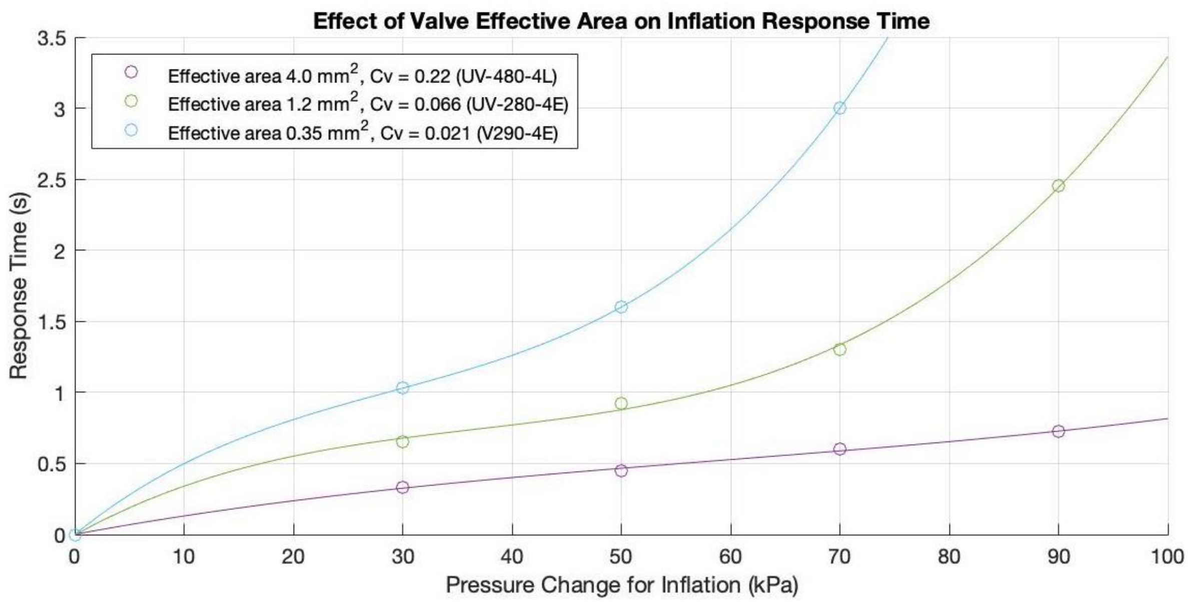
Pneumatic Circuit Name	Code	Valve 1	Valve 2
V1	100000000	UV-480-4L • 6 mm tubes	UV-480-4L • 6 mm tubes
V2	100000000	UV-280-4E • 6 mm tubes	UV-280-4E • 6 mm tubes
V3	100000000	V290-4E • 6 mm tubes	V290-4E • 6 mm tubes
V4	100000000	UV-480-4L • 4 mm tubes	UV-480-4L • 4 mm tubes
V5	100000000	KV-190-4E • 4 mm tubes	KV-190-4E • 4 mm tubes
V6	100000000	UV-280-4E • 4 mm tubes	UV-280-4E • 4 mm tubes
V7	100000000	V290-4E • 4 mm tubes	V290-4E • 4 mm tubes

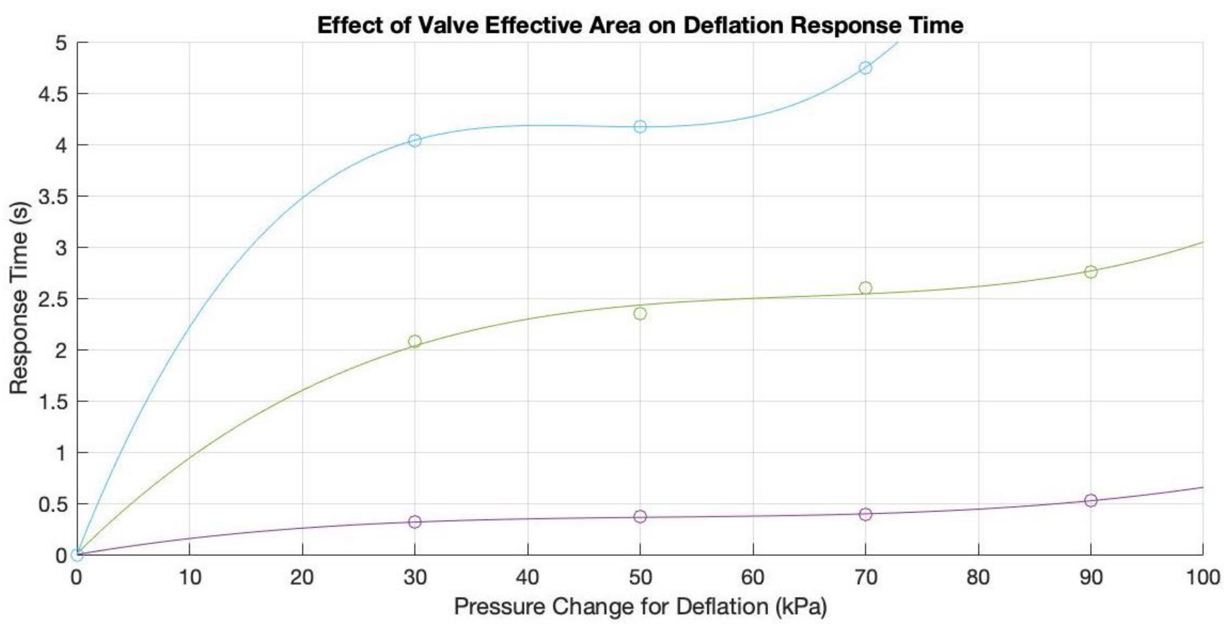
Multiple ports experiment:

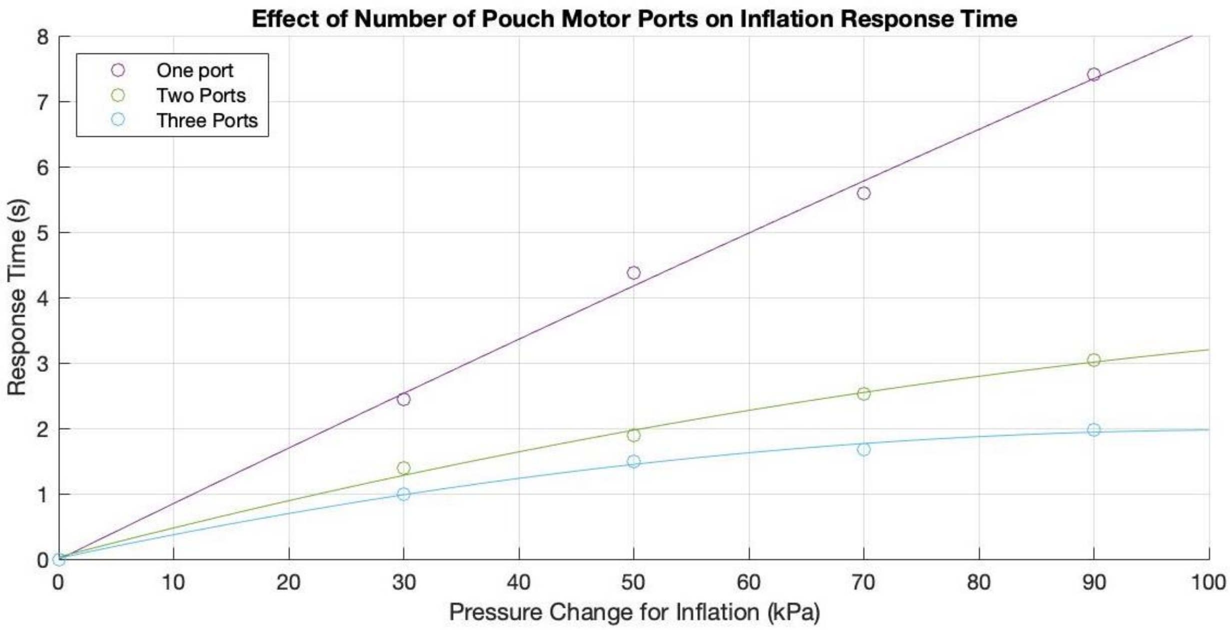


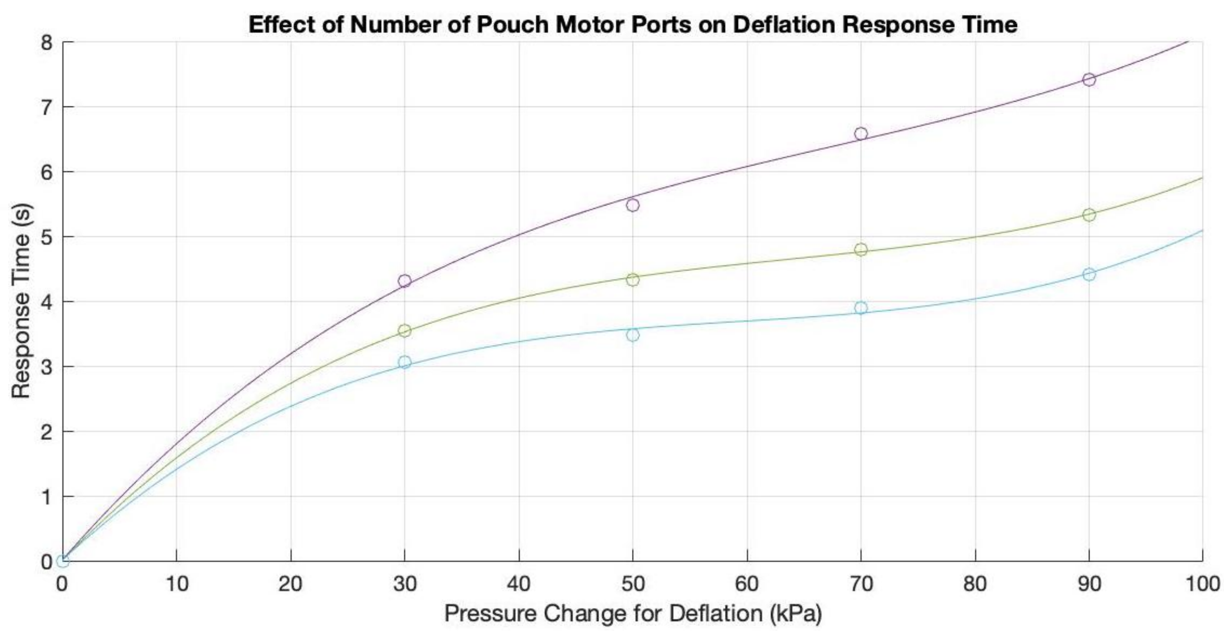
Pneumatic Circuit Name	Code	Valve 1	Valve 2	Valve 3	Valve 4	Valve 5	Valve 6
P1	110000000	UV 280-4E (6 mm tubes)	V290-4E (6 mm tubes)	N/A	N/A	N/A	N/A
P2	110000000	UV 280-4E (6 mm tubes)	KV-190-4E (4 mm tubes)	N/A	N/A	N/A	N/A
P3	111100000	UV 280-4E (6 mm tubes)	V290-4E (6 mm tubes)	UV 280-4E (6 mm tubes)	V290-4E (6 mm tubes)	N/A	N/A
P4	111100000	UV 280-4E (6 mm tubes)	KV-190-4E (4 mm tubes)	UV 280-4E (6 mm tubes)	KV-190-4E (4 mm tubes)	N/A	N/A
P5	111111000	UV 280-4E (6 mm tubes)	V290-4E (6 mm tubes)	UV 280-4E (6 mm tubes)	V290-4E (6 mm tubes)	UV 280-4E (6 mm tubes)	V290-4E (6 mm tubes)
P6	111111000	UV 280-4E (6 mm tubes)	KV-190-4E (4 mm tubes)	UV 280-4E (6 mm tubes)	KV-190-4E (4 mm tubes)	UV 280-4E (6 mm tubes)	KV-190-4E (4 mm tubes)

Experimental Results:
https://docs.google.com/spreadsheets/d/1uE3q_BeV7O1gR2C3nMLrATh69Z7SU7Hz0MSHC0l3Biw/edit#gid=0

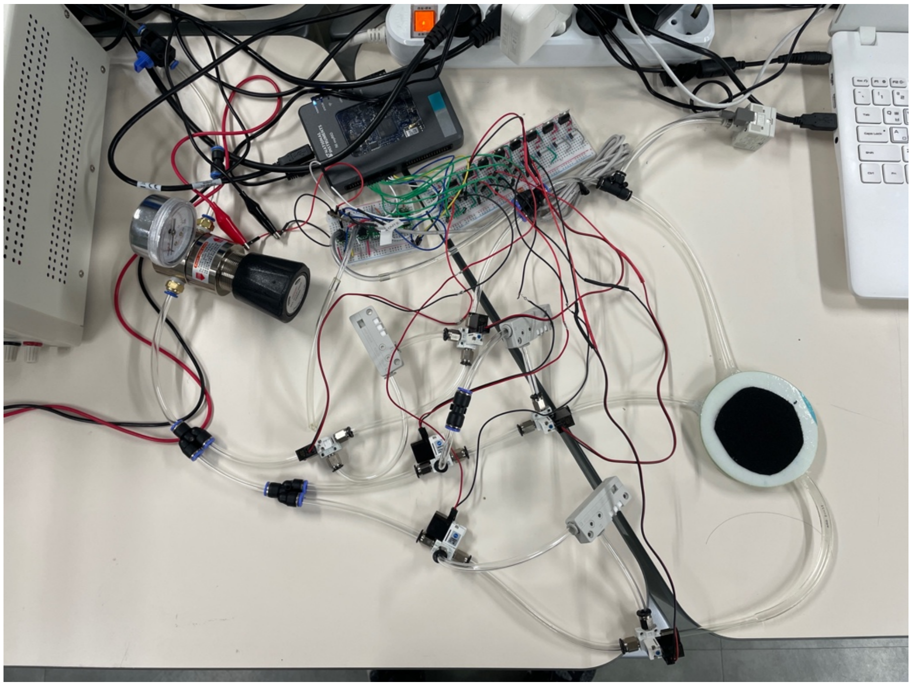




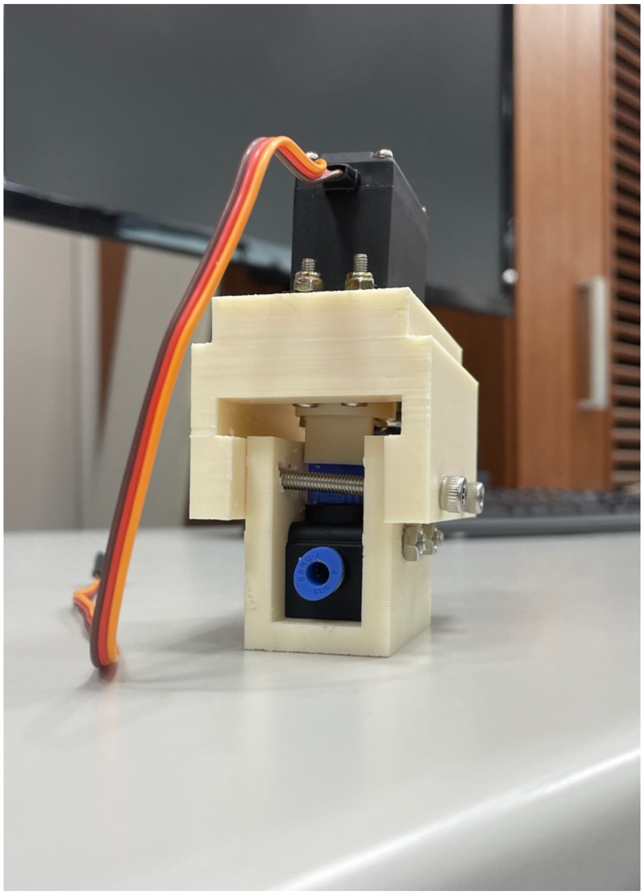

Effect of Number of Pouch Motor Ports on Inflation Response Time
One port
Two Ports
Three Ports
Response Time (s)
Pressure Change for Inflation (kPa)
0
1
2
3
4
5
6
7
8
0
10
20
30
40
50
60
70
80
90
100


Effect of Number of Pouch Motor Ports on Deflation Response Time
Response Time (s)
Pressure Change for Deflation (kPa)
0
1
2
3
4
5
6
7
8
0
10
20
30
40
50
60
70
80
90
100

Experimental Setup Pics:



Hand Valve Flow Controller:



Final To Do List:

- Finish CO_2 inflator design
- 3D print the CO_2 inflator
- Rewire pneumatic circuit using double 2/2 valve configurations
- Write pneumatic PWM PID controller on LabVIEW
- Test new pneumatic circuit with new PID controller
- Write pneumatic PWM bang bang controller on LabVIEW
- Test new pneumatic circuit with new bang bang controller
- Compare these results with previous bang bang controller with 3/2 solenoid valves